Sommario

RUBRICHE

Direttore: Alessandro de Simone

Redazione/collaboratori: Claudio Baiocchi, Carlo e Lorenzo Barazzetta, Giovanni Bellù, Simone Bettola, Andrea e Alberto Boriani, Diego e Federico Canetta, Giancarlo Castagna, Umberto Colapicchioni, Pasquale D'Andreti, Maurizio Dell'Abate, Valerio Ferri, Luca Galluzzi, Michele Maggi, Giancarlo Mariani, Marco Miotti, Flavio Molinari, Claudio Mueller, Massimo Pollutri, Carla Rampi, Fabio Sorgato, Giovanni Verrelli, Antonio Visconti.

Segreteria di redazione: Maura Ceccaroli.

Ufficio Grafico: Arturo Ciaglia

Direzione, redazione, pubblicità: V.le Famagosta, 75 - 20142 Milano - Tel. 02/8467348

Pubblicità: Milano: Leandro Nencioni (direttore vendite), Giorgio Ruffoni, Roberto Sghirinzetti Claudio Tidone - V.le Famagosta, 75 - 20142 Milano - Tel. 02/8467348

- Emilia Romagna: Spazio E - P.zza Roosevelt, 4 - 40123 Bologna - Tel. 051/236979
- Toscana, Marche, Umbria: Mercurio Srl - via Rodari, 9 - San Giovanni Valdarno (Ar) - Tel. 055/947444
- Lazio, Campania: Spazio Nuovo - via P. Foscari 70 - 00139 Roma - Tel. 06/8109679

Segretaria: Marina Vantini - **Abbonamenti:** Paola Bertolotti

Tariffe: prezzo per copia L. 3.500. Abbonamento annuo (11 fascicoli) L. 35.000. Estero: il doppio. Abbonamento cumulativo alle riviste Computer e Commodore Computer Club L. 70.000. I versamenti vanno indirizzati a: Systems Editoriale Srl mediante assegno bancario o utilizzando il c/c postale n. 37952207

Composizioni: Systems Editoriale Srl - **Fotolito:** Systems Editoriale Srl

Stampa: La Litografica S.r.l. - Busto Arsizio (VA)

Registrazione: Tribunale di Milano n. 370 del 2/10/82 - Direttore Responsabile: Michele Di Pisa

Sped. in abb. post. gr. III - Pubblicità inferiore al 70% - **Distrib:** MePe, via G. Carcano 32 - Milano

A tutto disco

Da un po' di tempo C.C.C. privilegia gli utenti del floppy disk perchè...

di Alessandro de Simone

Avevamo promesso di tener conto dei risultati dell'inchiesta del N.33. Benchè non sia stata più citata, i lettori più attenti avranno sicuramente notato alcuni cambiamenti in vari settori della nostra rivista.

Alcune rubriche sono state potenziate, altre sono state... sospese, altre ancora sono risorte a nuova vita.

Pur non potendo rivelare per intero i risultati dell'inchiesta, possiamo assicurare che abbiamo tenuto in gran conto i desideri dei lettori.

Abbiamo, ovviamente, iniziato ad accontentare subito le richieste sollecitate delle percenuali più alte, mentre per ciò che riguarda le altre (pur se consistenti) ci stiamo organizzando: possiamo anticipare fin da ora che il prezzo di copertina sarà ritoccato a partire dal prossimo numero (L.4000 invece di 3500) per consentire l'incremento di qualità e di pagine richiesto a gran voce dal 92% dei lettori. Coloro che possiedono un Plus/4 o un C/16 avranno una rubrica tutta per loro mentre chi possiede un C/128 troverà di certo il modo di utilizzarlo anche in modo diverso dal solito C/64.

Ma chi accontentiamo per primi? Quali argomenti affrontare subito? Quale è "il grido di dolore" pressante che abbiamo tratto dai risultati dell'inchiesta?

Ma è ovvio: i possessori di disk drive sono ormai talmente numerosi da consentirci di proporre una rubrica come "Directory". E in questo numero, oltre a Directory N.2, troverete un gruppo di programmi ed utility decisamente interessanti per i possessori del disk drive.

E tu, che sei rimasto nella bassa percentuale dei possessori del solo registratore, che aspetti a procurarti un bel 1541 per spiccare il tanto sospirato "salto di qualità"?

Natale è vicino...

Avvertiamo i lettori che intendono collaborare di attenersi RIGOROSAMENTE alle seguenti disposizioni:

- **NON inviare programmi e/o articoli che non siano stati espressamente concordati con la Redazione.**
- **Per risolvere il problema precedente, telefonare in Redazione (02/84.67.34.8) per proporre la collaborazione.**
- **Una volta concordato il "tema", e portato a termine il programma e l'articolo, inviarli su SUPPORTO MAGNETICO; non abbiamo il tempo di digitare nè l'uno nè l'altro.**
- **Per scrivere l'articolo servirsi di un Word Processor commercializzato: Easy Script C/64, Superscript C/128, Word Pro 3, Wordcraft Vic 20, W/P del Plus/4, Magic Desk o altri concordati nel corso della suddetta telefonata.**
- **I lavori vengono compensati (a seconda della qualità, interesse e rispondenza a quanto concordato) da un minimo di 25000 ad un massimo di 70000 lire per ogni pagina pubblicata. Esempio: articolo e programma che occupano 4 pagine della rivista: compenso variabile tra 100000 e 280000 lire.**

Rispettando le disposizioni eviterete perdite di tempo, di denaro (per spedizioni non richieste) e di pazienza.

Concatenare file sequenziali

☐ **Vorrei ulteriori informazioni sull'istruzione "Append" per quanto riguarda la concatenazione di file sequenziali tra loro.**

(Carlo Zardetto, Busto Arsizio)

• Precisiamo, anzitutto, che la concatenazione di più file sequenziali si può attuare soltanto con il drive e non con il registratore per motivi di Hardware sui quali non ci intratteniamo.

Supponiamo di dover registrare su disco un file sequenziale di nome "Prova" formato da una decina di numeri. Il programma può essere il seguente:

```
100 OPEN8,8,8,"PROVA,S,W"
110 FORI=1TO10
120 PRINT#8,I
130 NEXT:CLOSE8
```

Se carichi la Directory, dopo che il Led del drive si spegne, potrai notare la presenza, appunto, del file "Prova" che occupa un solo blocco.

Volendo leggere il contenuto del file appena registrato, potrai servirti del prossimo programma:

```
100 OPEN8,8,8,"PROVA,S,R"
110 INPUT#8,I
120 IF ST THEN CLOSE8:END
130 PRINTI:GOTO110
```

Il controllo sulla variabile ST, come saprai, serve per controllare se, in fase di lettura, si è giunti al termine del file.

Supponiamo, ora, di voler aggiungere altri numeri allo stesso file "Prova". Non sarà necessario, come si potrebbe pensare a prima vista, caricare in memoria il contenuto dell'intero file, aggiungere gli altri valori desiderati e, in seguito, scaricare nuovamente su disco il nuovo file.

Sarà invece sufficiente utlizzare il seguente programma:

```
100 OPEN8,8,8,"PROVA,S,A"
110 FORI=100TO110
120 PRINT#8,I
130 NEXT:CLOSE8
```

in cui, appunto, il carattere "A" presente tra virgolette, sta a significare "Append". Esaminando ora la directory ti accorgerai che il file di prima si è ingrossato (occupando 2 blocchi) e, leggendolo di nuovo col programma N.2, noterai che è formato da venti valori: da 1 a 10 del primo programma e da 100 a 110 del terzo.

In questo modo è possibile riempire un intero dischetto con un solo file sequenziale formato da concatenazioni attuate in più fasi successive.

Basic in Easy Script

☐ **Lavorando con il programma di Word Processor "Easy Script" mi capita di dover inserire nel testo programmi Basic da commentare. Finora ho sempre fatto ricorso ad un "Collage" e a successive fotocopie per migliorare l'aspetto finale.**

C'è un sistema per caricare da Easy Script un listato Basic?

• Certo: supponiamo che il programma Basic abbia il nome "Prova". Prima di usare Easy Script dovrai seguire una procedura semplicissima:

1/ Caricare il programma "Prova"

2/ Digitare, su di una stessa riga, i seguenti comandi in modo diretto:
OPEN 8,8,8, "NOME,S,W":
CMD 8: LIST
A questo punto il drive ronza e, dopo un tempo dipendente dalla lunghezza del programma, ricompare il cursore lampeggiante.

3/ Digita, quindi:
PRINT#8:CLOSE8

4/ Carica Easy Script e, una volta attivo, esegui Load "Nome". Il programma, che è stato trasformato in file sequenziale, viene letto senza alcun problema. In coda troverai sempre una strana parola (jready) che potrai cancellare con la funzione Delete (F1+ "D").

Da notare che il programma viene caricato in forma minuscolo maiuscolo (basta però usare F1+"U").

I caratteri speciali (Crsr. Home. Colori eccetera) che, di norma, nei listati Basic sono visualizzati con lettere maiuscole in Reverse, vengono però riportati tal quali, ma in modo "normale". Dovrai, pertanto, modificare il programma Basic ricorrendo ai caratteri CHR$.

Per cancellare lo schermo, ad esempio, non ricorrere al cuoricino tra virgolette, ma utilizza PRINT CHR$(147).

Bug in Tot/13

Per un errore in fase di composizione, l'ultima riga basic (1:END) di pag.27 del numero 35 di C.C.C. (che sembra relativa al programma Dispay per Tot/13) deve essere invece inserita al termine della riga 770 del programma Tot/13; questa, quindi, diventa:

770 PRINT#1,PEEK(J): NEXT: CLOSE1: END

Ci scusiamo con i lettori per l'involontario inconveniente.

La rivista "Commodore"

□ Non riesco più a trovare in edicola la vostra rivista "Commodore". Ne avete cessato la publicazione?

(Andrea Piemonte, Avellino)

• Da diverso tempo le tre riviste che prima pubblicavamo separatamente (Commodore, MSX, Sinclair Computer) sono ancora in edicola, ma "fuse" insieme nella rivista "Personal Computer" della Systems editoriale.

La testata "Commodore", però, la rivedrai presto in edicola dal momento che abbiamo deciso di utilizzarla per proporre argomenti monografici di grande interesse. Occhio all'edicola, dunque, per non farti sfuggire il prossimo fascicolo specialissimo sul Linguaggio Macchina applicato alla grafica del C/64!

La variabile ST

□ A che cosa serve la variabile riservata ST?

(Andrea Marchioro, Vicenza)

• Serve a controllare la correttezza delle operazioni di lettura dei file. Supponiamo di creare un file mediante il seguente programma...

```
100 OPEN1,8,8,"PROVA,S,W"
110 FORI=1TO10
120 PRINT#1,"PIPPO"
130 NEXT:CLOSE1
```

...e supponiamo che, in seguito, dimentichiamo il contenuto del file. Per leggerlo possiamo comunque utilizzare quest'altro listato:

```
100 OPEN1,8,8,"PROVA,S,R"
110 GET#1,X$:PRINTX$;
120 GOTO110
```

Poichè, infatti, non sappiamo nè il numero dei caratteri nè la loro natura, non possiamo ricorrere ad un ciclo For...Next. Se fai girare il programma, però, ti accorgi che, dopo la lettura dei dieci valori, compaiono segni "strani" sul video e il drive gira a vuoto senza fermarsi: non è presente, infatti, un controllo sulla fine del file. Per ovviare all'inconveniente è sufficiente modificare il listato di prima come segue:

```
100 OPEN1,8,8,"PROVA,S,R"
110 PRINTST;
115 IF ST>0 THEN CLOSE 1:END
120 GET#1,X$:PRINTX$;
130 IFX$=CHR$(13)THEN110
140 GOTO120
```

In conclusione, la variabile ST, letta dopo ogni operazione di lettura, deve essere sempre eguale a zero. Se, invece, assume altri valori vuol dire che si è verificato uno o più "inconvenienti". Ecco, di seguito, il codice di alcuni valori di ST ed il loro significato:

1, 2 Errori di temporizzazione

4 Blocco corto

8 Blocco lungo

32 Checksum Error

64 Fine del File

-128 Fine del nastro

Vivere pericolosamente

□ Avevo il programma "Print Shop" finchè non mi è venuto il prurito di cambiar nome al disco, e da allora non funziona più. Se, come penso, ho cancellato qualcosa di importante, come fare per ripristinarne il normale funzionamento?

(Raffaele Fittaiolo, Taranto)

• Quasi certamente hai combinato un bel guaio dal momento che "Print Shop", sia in versione originale che copiata, possiede al suo interno una serie di provvedimenti anticopia che effettuano alcuni controlli in varie parti del disco per accertarsi che il floppy non sia una copia pirata.

Nonostante questi provvedimenti siano del tutto inutili (le copie pirata, infatti, circolano in numero maggiore rispetto alle originali) è indispensabile non alterare per nessun motivo tracce e settori dei dischetti, nemmeno per operazioni "innocenti" come il cambio del nome. Nel tuo caso, purtroppo, non c'è nulla da fare, se non procurarsi un altro esemplare di dischetto funzionante.

Font di caratteri

□ Esiste un programma che consente di cambiare i tipi di carattere (font) della stampante MPS 803?

(Mario Cosma, Venezia)

• E' possibile, a patto di programmare, uno per uno, i caratteri desiderati. A dispetto della fatica improba, però, alla fine del lavoro otterresti una utility lentissima e di scarsa applicazione pratica.

L'unico modo per aggirare l'ostacolo sarebbe quello di sostituire la ROM dei caratteri della stampante con una Eprom personalizzata secondo i propri gusti.

L'operazione non è del tutto assurda tanto che, in più occasioni, ho proposto ai responsabili di numerose Ditte di realizzare, e commercializzare, le suddette ROM.

Finora nessuno ha accolto l'invito, forse perchè l'operazione non è considerata redditizia nonostante l'elevato numero di utenti interessati.

C'è qualche lettore eperto di hardware che vuole cimentarsi nell'impresa? Pubblicheremmo volentieri il suo lavoro e non avremmo difficoltà a indicare il suo nominativo per la vendita per corrispondenza di Eprom e accessori vari...

il joystick anatomico, per ogni tipo di mano...

- design rivoluzionario, si adatta perfettamente alla tua mano
- dotato di micro-switch svizzeri
- sopporta fino a 10 milioni di sollecitazioni

Mastertronic/Soft Center Via Mazzini, 15 Casciago (VA) Tel. 0332/212255

QUALSIASI COMPUTER

Scomposizione in fattori primi

Un listato che si rivela molto utile quando non si vuole perder tempo a scomporre i numeri dettati dal professore di matematica

di Maurizio Dell'Abate

Non stiamo a spiegare che cosa sia una scomposizione in fattori primi; i diretti interessati (gli studenti delle medie e, talvolta, delle superiori) lo sanno benissimo (!).

Grazie a questo programma tutto è reso automatico e a prova di errore: non resta, in seguito, che riportare sul quaderno la comoda videata.

N.B. Nei DATA sono contenuti i fattori primi con i quali può operare il programma. Tali valori sono più che sufficienti per numeri di bassa e media grandezza, ma quanti volessero potenziare il listato possono aggiungerne a volontà ricavandoli da una delle tabelle spesso presenti nelle appendici dei testi di matematica.

I più raffinati possono provvedere a compilare il programma che, in certi casi, richiede tempi lunghi di elaborazione.

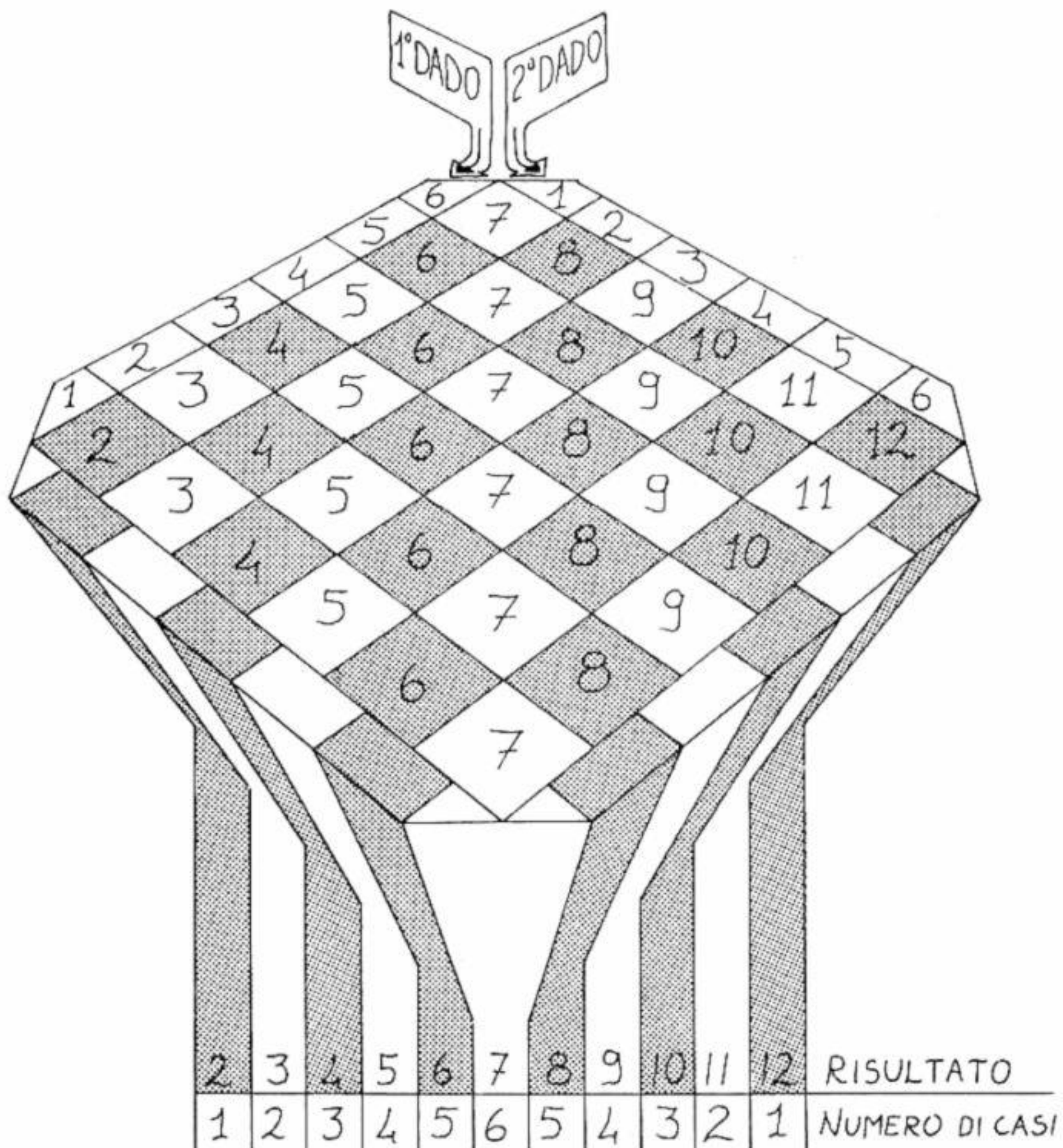
```
100 REM SCOMPOSIZIONE IN FATTOR
    I PRIMI
110 REM BY MAURIZIO DELL'ABATE
120 :
130 PRINTCHR$(147)
140 PRINT:PRINT TAB(4);CHR$(18)
    ;" PROGRAMMA PER LA SCOMPOS
    IZIONE "
150 PRINT TAB(2);CHR$(18);" IN
    FATTORI PRIMI DI NUMERI INT
    ERI "
160 PRINT:PRINT:PRINT
170 INPUT "NUMERO";A:A=INT(ABS(
    A)):IF A<=1 THEN  RUN
180 PV=1
190 PRINTCHR$(147):PRINT:PRINT"
         ";A;:REM 5SPAZI
200 RESTORE :FOR Q=1 TO PV:READ
     DV:IF DV>0 THEN NEXT:GOTO
    220
210 GOTO 290
220 ZX=A/DV:IF ZX<>INT(ZX) THEN
     PV=PV+1:GOTO 200
230 A=ZX:PRINT TAB(15);CHR$(221
    );"  ";DV:REM 2 SPAZI
240 PRINT"     ";A;:REM 5SPAZI
250 IF A<>1 THEN 200
260 PRINT:PRINT TAB(15);"PREMI
    UN TASTO"
270 GET W$:IF W$="" THEN 270
280  RUN
290 PRINTCHR$(147)
300 PRINT" PER SCOMPORRE QUESTO
     NUMERO, OCCORRE"
310 PRINT" AGGIUNGERE ALTRI NUM
    ERI PRIMI NEI"
320 PRINT" DATA IN CODA AL PROG
    RAMMA."
330 GOTO 260
340 :
350 REM NUMERI PRIMI
360 DATA 2,3,5,7,11,13,17,19,23
    ,29,31,37,41,43,47,53,59,61
    ,67,71,73,79,83,89
370 DATA 97,101,103,107,109,113
    ,127,131,127,131,137,139,14
    9,151,157,163,167
380 DATA 173,179,181,191,193,19
    7,199,211,223,227,229,233,2
    39,241,251,257,263
390 DATA 269,271,277,281,283,29
    3,307,311,313,317,331,337,3
    47,349,353,359,367
400 DATA -1:REM  END FLAG
```

GIOCHI D'AZZARDO

QUALSIASI COMMODORE

La tassa sugli imbecilli

Viene descritto in che modo lo Stato (con il Lotto) e la Camorra (Lotto clandestino) riescono sempre a vincere montagne di denaro grazie al popolare gioco del sabato

di Claudio Baiocchi

Se, all'inizio di una tombolata, un amico vi proponesse "Scommetto 1000 lire che il primo numero estratto è dispari", si tratterebbe di una scommessa che i matematici chiamano EQUA: dei 90 casi possibili, infatti, 45 sono a vostro favore e 45 sono contro.

Se invece vi venisse proposto: "1000 lire che il primo numero sarà minore di 50" l'amico sta cercando il pollo: lui ha, infatti, 49 possibilità a favore, voi solo 41. La scommessa sarebbe "equa" se, invece che "alla pari", fosse formulata "49 contro 41". Esempio: "Se esce un numero da 50 in su, ti pago 4900 lire, altrimenti me ne dai 4100".

Infine un amico che accettasse scommesse sul primo estratto, dicendo: "Se sbagliate a indovinare mi date 1000 lire, se indovinate vi pago io 100 mila" va portato al manicomio: di fronte a 90 scommesse, ognuna su un numero diverso, si troverebbe ad incassare 89 volte 1000 lire, per pagarne poi 100 mila.

Dalla considerazione precedente si è portati a concludere che la scommessa sul primo estratto è equa se fatta "89 contro 1" ovvero, in caso di vincita, si riscuote 90 volte la posta. Qui la conclusione è corretta, però questo tipo di ragionamenti, in casi leggermente diversi, può portare a risultati sbagliati (si dia un'occhiata alla figura).

D'altro lato molte persone sarebbero disposte ad accettare scommesse del tipo "62 contro 1" sul primo estratto, in base alla considerazione:

"Se perdo, neanche mi accorgo di avere 1000 lire in meno; invece, se vinco, con 62 mila lire si fa già una bella cenetta..."

Si tratta di un ragionamento psicologicamente ineccepibile, ma in realtà, accettando la scommessa, rimarreste vittima di un furto a causa della scommessa "iniqua".

Truffe o opere di bene?

La quota "62 contro 1" non l'ho scelta a caso: è effettivamente la scommessa che lo Stato propone ogni settimana a chiunque voglia giocare sul "primo estratto" al gioco del Lotto. In effetti l'indicazione sul tagliando è "63 volte la posta". Giocando (e pagando in anticipo) 1000 lire, in caso di vincita ne incasserete 63000, per un utile netto di 62000 (invece delle 89000 che teoricamente renderebbero il gioco equo).

E' un furto? Beh, mettetevi nei panni dello Stato: con le somme giocate deve stipendiare i dipendenti dei Botteghini del Banco Lotto sparsi in tutta Italia, quindi una "tassa" di 27000 lire sulle 89000 teoriche può sembrare più che ragionevole (o no?). Certo è, che se tentaste di tener voi banco ad un gioco del genere, probabilmente finireste dentro per bisca clandestina.

Per altri tipi di scommessa, come quella sulla cinquina, l'entità del prelievo statale effettuato sulle vincite risulta tanto scandalosa (listato 3) che, fin dal nascere, il Lotto fu soprannominato "la tassa sugli imbecilli".

I ritardi

E veniamo al controverso problema dei "numeri che ritardano". Mentre sto preparando questo articolo (settembre '86), una pagina del TELEVIDEO destinata al problema dei numeri ritardatari informa che, sulla ruota di Torino, il numero 70 non esce da ormai 101 settimane.

E' corretto dedurre da tale informazione che è un investimento vantaggioso puntare grosse cifre sul 70 "estratto semplice" per la ruota di Torino?

E magari aumentare la posta nelle settimane successive, qualora il 70 continuasse a ritardare?

Dal punto di vista matematico la risposta è negativa, così come è negativa la risposta all'analogo problema posto dal giocatore che tenta di fare i soldi alla Roulette puntando "al raddoppio" su Rosso-Nero.

Per capirne il perchè, è bene cominciare da un gioco più semplice del Lotto, che è poi molto simile al Rosso-Nero della Roulette (dove però la presenza dello zero è un handicap in più): giocheremo cioè con una moneta "ben equilibrata" e col problema del "Testa o Croce".

In questo gioco un problema sui ritardi potrebbe essere così formulato: avete assistito a qualche lancio di una moneta, ed avete visto uscire TESTA per sette volte di fila; venite invitati a scommettere sul prossimo lancio; cosa scegliete?

Indipendentemente dalla risposta che avete in mente, sospendete per un attimo la riflessione su questo problema e seguitemi nell'organizzazione del seguente:

PRIMO TEST. Supponete di dare ad otto persone una moneta ben equilibrata; invitate ognuna di esse ad effettuare un lancio; poi chiedete ad ognuno il risultato ed annotatelo su un foglio di carta.

Quanti sono i possibili risultati? Il conto è facile: la prima risposta fornisce due eventualità, ognuna delle quali si ramifica in due quando si aggiunge la seconda risposta. Fin qui i casi sono 4: TT, TC, CT, CC, avendo abbreviato Testa con "T" e Croce con "C"; ognuno dei 4 casi ne genera altri due quando si tiene conto della terza risposta, e così via per le altre.

Si tratta perciò di eseguire 2*2*...*2 otto volte; cioè 256.

Se invece delle lettere "T" e "C" annotate i risultati scrivendo 1 per il risultato Testa e 0 per il risultato Croce, un "esperimento" corrisponde ad una serie di otto cifre binarie. Tradotto il numero in decimale, ogni esperimento corrisponde a uno dei numeri 0, 1, 2, 3, ...,255: e viceversa ogni numero tra 0 e 255, tradotto in binario (e con l'eventuale aggiunta di qualche 0 iniziale) corrisponde ad un possibile risultato dell'esperimento: 256 possibili risultati, ognuno rappresentato con un valore compreso tra 0 e 255.

Se si è d'accordo sul fatto che i singoli risultati ottenuti dalle otto persone sono tra di loro indipendenti (e come potrebbe il lancio effettuato dal primo influire su quello effettuato dal secondo, o dal terzo?) il nostro primo test è da considerarsi completamente equivalente al seguente:

SECONDO TEST. Si sceglie a caso un numero intero tra 0 e 255.

Il passo successivo, sul quale il matematico non ha dubbi, ma che è spesso rifiutato da chi preferisce la cabala alla matematica, è il seguente: il primo test (e quindi anche il secondo) è perfettamente equivalente al:

TERZO TEST. Una persona lancia otto volte una moneta, annotando via via i risultati dei vari lanci.

A questo punto il problema relativo al Testa e Croce proposto in precedenza può essere così riformulato: è stato estratto un numero a caso tra 0 e 255; si sa che questo numero è più grande di 253 (perchè, scritto in binario, le prime sette cifre sono tutte 1). Il prossimo sarà il 254 o il 255?

La risposta è ovvia: non ci sono motivi per scommettere sul 254 piuttosto che sul 255!

Naturalmente potete non essere d'accordo sulla equivalenza dei test considerati; però in tal caso è inutile andare a cercare sui vostri vecchi libri di Matematica un'idea per vincere al Lotto giocando sui numeri che ritardano: da questi libri, interpretando correttamente le informazioni contenute, potrete solo concludere che non conviene giocare al Lotto.

Sperando che queste considerazioni non vi abbiano choccato troppo, vi dico come reagirei io se fossi invitato a puntare su Testa o Croce: sapendo solo che gli ultimi 7 lanci hanno dato Testa, io sarei indotto a pensare di essere di fronte a una delle tre eventualità:

- la moneta è truccata;
- chi lancia la moneta ha una particolare abilità di prestidigitazione;
- qualcuno dei presenti possiede strane doti di telecinesi, e se ne sta servendo.

In tutti e tre i casi (ma a malincuore, perchè se siamo nel primo caso sarebbe mio interesse scommettere su Testa) io rifiuto di giocare: i giochi-truffa non mi piacciono!

Esame della figura

Se lanciate 600 volte un dado non vi aspetterete certo di ottenere esattamente cento lanci con esito 1, cento con esito 2 e così via. Se, però, l'1 è uscito solo una ventina di volte ed il 6 è uscito quasi 200 dovreste cominciare a sospettare che il dado sia truccato. Per confermare questa ipotesi basta provare ad eseguire 6000 lanci e controllare se ancora la "frequenza" dell'1 e quella del 6 restano molto lontane.

Pensate ora di lanciare due dadi: se effettuate 12000 lanci, quante volte vi aspettate di ottenere 7? Se la vostra risposta non è "circa 2000 volte" dovete rifletterci su ancora un po'. Avete, infatti, impostato il discorso partendo dalla considerazione che i casi possibili sono 11 (e cioè: la somma dei dadi fa 2; oppure fa 3; oppure 4... fino a 12); senza però tenere conto del fatto che i casi considerati hanno una diversa probabilità: dovrebbe bastare un attimo di riflessione per rendersi conto che ottenere un 7 è "più facile" che ottenere un 12, oppure un 2; la figura 1 indica la maniera corretta di impostare il problema.

In particolare, se venite invitati a scommettere sul numero che uscirà lanciando due dadi, non potete aspettarvi una scommessa del tipo "10 contro 1": l'unica cosa ragionevole, da parte di chi vi invita a giocare, è proporvelo a 5 contro 1; e se voi accettate dovete puntare sempre sul 7!

Tramite il listato 1, che simula il lancio di un dado, potete controllare la "casualità" della funzione RND implementata sul vostro computer; il listato 2 simula invece il lancio di DUE dadi. A commento dei listati, qualche problemino per i lettori più diligenti:

- Nel listato 1, linea 10, è definita una funzione col nome DICE (è il nome inglese per dado). Potete naturalmente sostituirlo con DADO; ma perchè questa sostituzione genera un SYNTAX ERROR su C/16, PLUS/4 e C/128?
- Sempre nel listato 1, interpretate bene le REM in linee 20, 30.
- Problema analogo per il listato 2
- Nel listato 2, linea 10, l'espressione
INT(6*RND(1))+INT(6*RND(1))
non può essere sostituita con una delle espressioni (più semplici, ma che non danno il risultato voluto):
2*INT(6*RND(1))
oppure
6*RND(1)+6*RND(1)
Ne vedete la ragione?
- Modificate i listati 1 e 2, sopprimendo in linea 10 l'addendo 1 e 2 rispettivamente (è sempre un problema di velocità: l'operazione considerata va iterata un gran numero di volte...).

Notizie sul Lotto

Le estrazioni del gioco del Lotto si svolgono ogni sabato pomeriggio in 10 città italiane (BO, CA, FI, GE, MI, NA, PA, ROMA, TO, VE).

Alla presenza di un funzionario dell'Intendenza di Finanza, che garantisce la correttezza delle operazioni, da un'urna contenente 90 bussolotti numerati da 1 a 90 vengono estratti 5 numeri che, nell'ordine di estrazione, vengono poi comunicati agli organi di informazione.

Per facilitare il rimescolamento dei numeri le urne avevano anticamente la forma di una ruota: una giocata fatta "per la ruota di Napoli" significa semplicemente che la scommessa concerne solo i numeri che usciranno nell'estrazione svolta a Napoli. Una giocata di 9000 lire, fatta "per tutte le ruote", equivale a 10 giocate (tutte da 900 lire), fatte ognuna per una ruota.

Oltre al "primo estratto", di cui abbiamo già parlato nel corso dell'articolo, il gioco del Lotto prevede la possibilità di effettuare vari tipi di scommesse per una o più ruote (faremo riferimento ad un'unica ruota):

- "Estratto semplice": si scommette che un determinato numero sarà tra i 5 estratti sulla ruota prescelta.
- Ambo, terno, quaterna, cinquina: dei cinque numeri che risulteranno estratti, si pretende di indovinarne rispettivamente due, tre, quattro, o tutti e cinque.

Per calcolare la probabilità di vincita di queste giocate basta fare uso dei "coefficienti binomiali" (se li avete dimenticati, il listato 3 vi aiuterà).

Ad esempio esistono 90*89/2 = 4005 ambi diversi; ed ogni estrazione fornisce 5*4/2 = 10 ambi vincenti; il

gioco sarebbe matematicamente equo se lo stato pagasse 400 volte e mezza la posta (invece la quota stabilita dalla legge è di sole 250 volte la posta).

Le altre quote previste dalla legge sono:
10 volte e mezzo la posta per l'estratto semplice (ma si veda il prossimo paragrafo);
4000 volte la posta per il terno;
80000 volte la posta per la quaterna;
un milione di volte la posta per la cinquina.

Si osservi che la "tassa" automaticamente percepita dallo Stato sulle vincite è via via più elevata: sulla cinquina la "quota equa" che lo Stato dovrebbe pagare corrisponde ad oltre 43 milioni di volte la posta!

Il listato 3 calcola automaticamente la vincita corrispondente ad ogni puntata: l'apparente complicazione del listato è dovuta al fatto che, volendo puntare 10000 lire sull'uscita dei numeri 2, 55 e 90 per la ruota di Napoli, siete autorizzati a ripartire le vostre diecimila giocando "7000 sul terno e 3000 sull'ambo". In realtà sarebbe più corretto dire "3000 sugli ambi" perchè con tre numeri si possono formare TRE ambi (nell'esempio del terno 2, 55, 90: l'ambo 2-55, quello 2-90 e quello 55-90). Naturalmente la vostra puntata equivale ad una puntata di 7000 lire sul terno e di tre puntate, ognuna da 1000 lire, su ognuno degli ambi.

Una furbizia popolare

Se, entrati in un botteghino del Banco Lotto, provate a chiedere di giocare "89000 lire sul 35 per la ruota di Napoli", vi sentirete probabilmente domandare dall'impiegato, con tono meravigliato: "Per ambata?".

Si tratta di una giocata "non ufficiale", nata in seguito ad un grossolano errore del legislatore, all'atto di decidere le quote relative alle vincite: un ambo viene pagato 250 volte la posta, mentre l'estratto semplice è pagato 10 volte e mezzo. Supponete dunque di giocare 89000 lire sul 35 (la cifra è scelta apposta per fare meglio i conti). Se avete fortuna incasserete 89000*10.5=934500 lire.

Supponete ora di chiedere al malcapitato impiegato di compilare 89 bollette da 1000 lire ciascuna: una sull'ambo 1-35, una sull'ambo 2-35, una sul 3-35 e così via fino all'ambo 90-35 (si tratta solo di 89 ambi, perchè naturalmente l'ambo 35-35 va saltato).

In questo caso spenderete, come prima, 89000 lire ma, sempre nell'ipotesi che il 35 venga estratto, incasserete leggermente di più: l'uscita del 35 fa sì che 4 dei vostri ambi risultano vincenti; poichè ognuno di essi vale 250 volte la posta, le vostre giocate vi faranno guadagnare un milione...

La Storia non ci ha tramandato il nome del furbacchiotto che ha scoperto questo sistema basato, badate bene, non su di una bizzarria matematica ma sulla fondamentale iniquità del gioco.

A causa, probabilmente, delle proteste dei dipendenti che si trovavano a dover "scrivere troppo", e forse anche per risparmiare sulla carta (per ogni giocata occorre compilare un tagliando madre ed uno figlia) lo Stato ha ormai accettato la giocata "per ambata", che sta appunto a significare: "gioco su tutti gli 89 ambi fatti con...". La vincita corrispondente all'ambata è naturalmente 1000/89 volte la posta.

```
  1 REM LISTATO 1
  2 REM LANCIO DI UN DADO
  3 :
 10 DEF FNDICE(X)=1+INT(6*RND(1
    ))
 20 REM  PER ACCELERARE L'ESECU
    ZIONE RISCRIVERE LA 10 NELL
    A FORMA :
 30 REM  DEFFNDICE(X)=1+6*RND(1
    )
 40 PRINT"SIMULAZIONE DEL LANCI
    O DI UN DADO"
 50 INPUT "QUANTE PROVE";N
 60 FOR X=1 TO N
 70 Y=FNDICE(X)
 80 A(Y)=A(Y)+1
 90 NEXT
100 FOR X=1 TO 6
110 PRINT"IL RISULTATO"X"E' USC
    ITO"A(X)"VOLTE"
120 NEXT
130 PRINT"(OGNI RISULTATO ERA A
    TTESO"N/6"VOLTE)"
140 END
```

```
  1 REM SECONDO LISTATO
  2 REM LANCIO DI DUE DADI
  3 :
 10 DEF FNDADI(X)=2+INT(6*RND(1
    ))+INT(6*RND(1))
 20 REM  PER ACCELERARE L'ESECU
    ZIONE RISCRIVERE LA 10 NELL
    A FORMA :
 30 REM  DEFFNDADI(X)=2+6*RND(1
    )+INT(6*RND(1))
 40 PRINT"SIMULAZIONE DEL LANCI
    O DI DUE DADI"
 50 INPUT "QUANTE PROVE";N
 60 DIM A(12):FOR X=1 TO N
 70 Y=FNDADI(X)
 80 A(Y)=A(Y)+1
 90 NEXT:PRINT
100 PRINT"NUMERO/CHE E' USCITO/
```

```
    ERA ATTESO CIRCA":PRINT
110 FOR X=2 TO 12
120 READ A
130 PRINTX TAB(5)A(X)"VOLTE",IN
    T(.5+A*N/36)"VOLTE
140 NEXT
150 DATA 1,2,3,4,5,6,5,4,3,2,1
160 END
```

```
  1 REM TERZO LISTATO:
  2 REM CALCOLO DELLA
  3 REM VINCITA AL LOTTO
  4 :
 10 PRINTCHR$(147)CHR$(17)CHR$(
    18)SPC(5)" GIOCO DEL LOTTO
    ":PRINT
 20 DIM N$(5):FOR X=1 TO 5:READ
     N$(X):NEXT
 30 DATA ESTRATTO SEMPLICE,AMBO
    ,TERNO,QUATERNA,CINQUINA
 40 DIM QP(5):FOR X=1 TO 5:READ
     QP(X):NEXT
 50 DATA 10.5,250,4000,80000,10
    00000:REM  QE=QUOTA PAGATA
 60 DIM CB(5,5):FOR A=1 TO 5:FO
    R B=1 TO A:GOSUB 380:CB(A,B
    )=Q:NEXT:NEXT
 70 DIM QE(5),N(5):REM  QE=QUOT
    A EQUA; N=NUMERO DI POSSIBI
    LITA'
 80 FOR X=1 TO 5:A=90:B=X:GOSUB
     380:N(X)=Q:QE(X)=Q/CB(5,X)
    :NEXT
 90 PRINT"POSSO LAVORARE SU TUT
    TE LE RUOTE"
100 PRINT"O SU UN'UNICA RUOTA;"
    :PRINT
110 PRINT"QUANTI NUMERI VUOI GI
    OCARE";:INPUT N
120 PRINT"PER TUTTE LE RUOTE";:
    INPUT R$:R=10:REM  R=# RUOT
    E
130 R=10:IF R$<"S" THEN R=1
140 IF N>1 THEN 250
150 PRINT"LO GIOCHI COME PRIMO
    ESTRATTO";:INPUT A$:IF A$<"
    S" THEN 180
160 PRINT"QUANTO GIOCHI";:IF R=
    1 THEN PRINT" SU OGNI RUOTA
    ";
170 INPUT Q:Q=Q/R:QE=Q*90:QP=Q*
    63:GOTO 210
180 PRINT"ALLORA GIOCALO PER AM
    BATA"
190 PRINT"QUANTO GIOCHI";
200 INPUT Q:Q=Q/R:QE=Q*18:QP=Q*
    1000/89
210 IF R=10 THEN PRINT"PER OGNI
     RUOTA IN CUI IL NUMERO ESC
    E"
220 PRINTCHR$(18)"VINCERAI"INT(
    QP):PRINT"(LA QUOTA EQUA SA
    REBBE"QE")"
230 PRINT:INPUT "ALTRE GIOCATE"
    ;A$:IF A$>"NO" THEN 110
240 END
250 PRINT:FOR X=1 TO N:PRINT"QU
    ANTO GIOCHI SU "N$(X);:INPU
    T Q:Q=Q/R
260 Q(X)=Q/CB(N,X):REM  QUOTA E
    FFETTIVAMENTE GIOCATA SU OG
    NI COMBINAZIONE X
270 NEXT
280 FOR X=1 TO N:Q=0:FOR Y=1 TO
     X:Q=Q+Q(Y):NEXT:IF Q=0 THE
    N 370
290 PRINT:PRINT"SE I NUMERI CHE
     ESCONO SONO "CHR$(18)XCHR$
    (146)" VINCERAI"
300 IF R=10 THEN PRINT"(PER OGN
    I RUOTA IN CUI ESCONO"X"NUM
    ERI)":PRINT
310 QE=0:QP=0:FOR Y=1 TO X:IF Q
    (Y)=0 THEN 340
320 Q=Q(Y)*CB(X,Y):QE=QE+Q*QE(Y
    ):Q=Q*QP(Y):PRINTQ"PER LA G
    IOCATA SU "N$(Y)
330 QP=QP+Q
340 NEXT:PRINTCHR$(18)"IN TOTAL
    E VINCI"QP:PRINT"(LA QUOTA
    EQUA SAREBBE"QE")"
350 PRINT"PREMI UN TASTO"
360 GET A$:IF A$="" THEN 360
370 NEXT:PRINT:PRINT:GOTO 230
380 Q=1:FOR Z=1 TO B:Q=Q*(A+1-Z
    ):Q=Q/Z:NEXT:Q=INT(Q+.5):RE
    TURN
390 REM  RESTITUISCE Q=COEFFICI
    ENTE BINOMIALE DI A SU B
400 END
```

EDP USA ROMA '86

MOSTRA E SEMINARI

EDP USA

25-28 NOVEMBRE 1986

PALAZZO DEI CONGRESSI DI ROMA EUR

TRADIZIONE e NOVITÀ: una perfetta fusione per questa settima edizione di EDP USA ROMA, l'unica rassegna di informatica americana dedicata al mercato dell'Italia Centromeridionale. Oltre che da alcuni seminari di studio sulle più attuali tematiche del settore, quali "Reti Geografiche e Reti Locali", "Software per Intelligenza Artificiale", e "Personal Computer e CAD", la mostra sarà caratterizzata dalla presenza di una sezione interamente riservata alle società statunitensi produttrici di software, chiamata "SOFTWARE VILLAGE", presso la quale sarà presente anche una missione ufficiale inviata dal Governo americano. Il Centro Commerciale Americano, organizzatore della manifestazione, è a disposizione per fornire ulteriori informazioni su mostra e seminari.

CENTRO COMMERCIALE AMERICANO

VIA GATTAMELATA, 5 / 20149 MILANO
TELEFONO 46.96.451-2 / TELEX 330208 USIMC-I

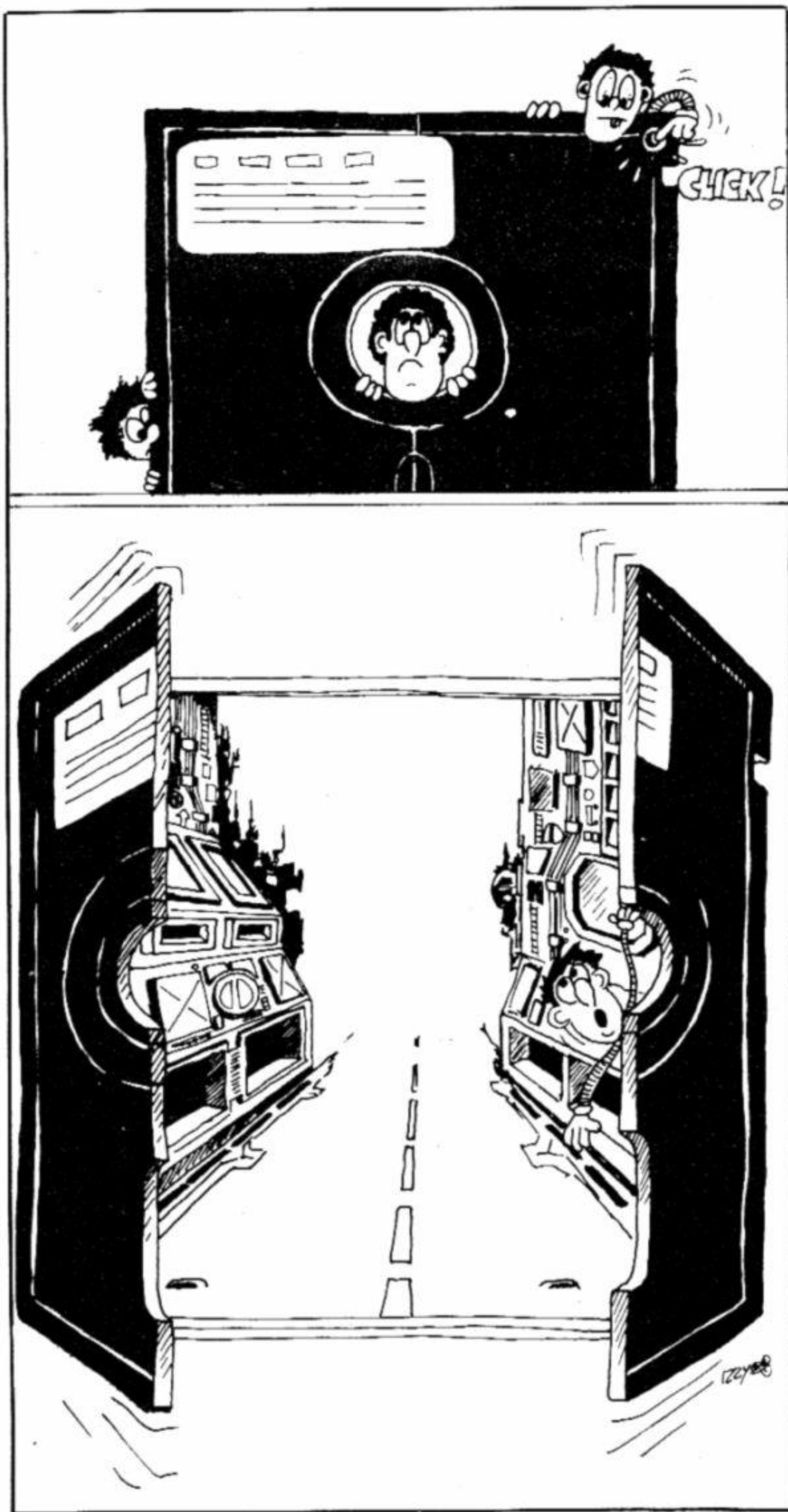
EDP USA 86

Che cosa c'è nei vostri floppy?

Una utility (in LM!) indispensabile per chi intende smanettare con il drive

di Paolo Agostini

Nessuno penserebbe di buttare giù un muro con le mani nude oppure di costruire una casa senza mattoni o cemento. Così è nel campo del computer: è impensabile fare qualcosa di valido se manchiamo degli strumenti necessari.

Uno di questi è proprio un "Monitor" per i floppy disk. Senza di questo non è possibile intervenire sui dischi per modificarne tracce o settori allo scopo di ottenere risultati un po' fuori della norma tra cui quello di proteggere, da intrusioni esterne, alcune tracce o determinati settori, o per mascherare la presenza di routine in linguaggio macchina nascoste in punti impensati del disco. O anche per studiare più a fondo il comportamento del DOS ed eventualmente trovare

gli artifizi che sono serviti ad altri per proteggere i loro programmi.

Il programma

Copiate con la massima attenzione il programma in BASIC. Questo listato, che chiameremo "codice sorgente" ha lo scopo di creare un "codice oggetto" su disco in modo da utilizzarlo, in seguito, con la massima velocità.

Ogni riga è composta di 10 dati più un dato di controllo. Il computer "somma" logicamente i dieci dati e li confronta, riga per riga, con la cifra di controllo. Se vi sono divergenze il programma si blocca e segnala le tre righe in cui è più probabile che abbiate commesso l'errore di trascrizione.

Se il computer si blocca senza segnalare l'errore in italiano (appariranno tipiche segnalazioni di errore in inglese), significa che avete scritto una cifra maggiore di 255 tra le righe di Data.

In questi casi, per rintracciare la riga dove avete commesso l'errore, è sufficiente dare in modo diretto il comando:

PRINT PEEK(63)+PEEK(64)*256

e sullo schermo apparirà l'ultima riga di dati "letta" dal computer. Se tutti i dati sono esatti, seguendo le istruzioni che appariranno su video verrà creato un programma (in linguaggio macchina), che sarà possibile caricare con un semplice LOAD"DISKMON",8 e far partire con RUN.

Come utilizzare il programma

Dopo, appunto, il Run di "Diskmon", vi troverete in ambiente monitor e apparirà la lista dei comandi a vostra disposizione. Esaminiamoli uno per uno.

$ (dollaro): questo comando trasforma 2 oppure 4 caratteri esadecimali in un numero decimale. Provate a dare il comando $C000, dovrebbe apparire sullo schermo la cifra #49152. Provate a dare il comando $12, sullo schermo dovrebbe apparire #18. Se introdurrete valori esadecimali di cinque o più cifre, saranno sempre le ultime 4 cifre introdotte ad essere trasformate in un numero decimale.

(cancelletto): questo comando trasforma un numero intero, (valore, cioè, senza virgole decimali) in un numero esadecimale. Il valore massimo consentito è di #65535. Se introdurrete numeri maggiori, apparirà un esadecimale di quattro cifre, ma si tratterà di un valore completamente casuale. Provate a dare il comando #13: dovrebbe apparire sullo schermo il valore esadecimale $000D.

X: questo comando consente di uscire dal Monitor (è l'abbreviazione di "eXit", usata in quasi tutti i monitor in l.m.)

Nel digitare righe di programma basic che contengono istruzioni DATA, è piuttosto facile incorrere in errori di digitazione. Supponiamo che un'ipotetica linea basic numerata con 1200 debba contenere i tre valori: 123, 456, 789. Ecco alcuni esempi di errori più frequentemente commessi:

1200 DATA,123,456,789

C'è una virgola dopo la parola "DATA". I dati letti dal computer sono, in questo caso, quattro: 0, 123, 456, 789. Se, infatti, non figura alcun carattere dopo l'istruzione DATA, automaticamente viene assunto il valore nullo (0).

1200 DATA 123,456,789,

In questo caso, dopo il numero 789, il computer, grazie alla presenza della virgola erroneamente inserita, "crede" che ci sia un altro valore e, non trovandolo, lo assume come nullo (0).

1200 DATA 1234,56,789

La virgola è posizionata male, vale a dire dopo il carattere "4" e non dopo il carattere "3". Il computer non può sapere se il valore esatto è 123 oppure 1234 e individuare un errore, in questo caso, risulta piuttosto laborioso.

At (=Chiocciolina, tasto che si trova tra la lettera "P" e l'asterisco "*"): si tratta di un comando molto potente, dal momento che consente di trasmettere ordini al DOS (Disk Operating System):

Il simbolo di chiocciolina, seguito da un semplice RETURN, leggerà il canale di errore del disk drive e farà apparire sullo schermo il messaggio 0,OK,0,0 oppure il messaggio di errore.

Digitando come PRIMO carattere quello della chiocciolina e come SECONDO uno dei seguenti, si ottengono diversi effetti:

$ visualizzerà sullo schermo la directory.
V convalida il disco.
I inizializza il disco.
S:nome file, cancellerà il file indicato.
R:nuovo nome file=vecchio nome file, cambierà il nome del file.
N:nome disco, effettua la "formattazione corta" di un disco già formattato, provvedendo a cancellarne la directory.
N:nome disco,ID effettua la consueta formattazione di un disco vergine.
UJ, resetterà il disk drive (corrisponde, per il disk drive, a quello che è per il Commodore 64 il comando SYS 64738).

Quattro funzioni fondamentali

E veniamo alle quattro funzioni più importanti del monitor.

Funzione READ: il comando deve essere dato come R seguito da uno spazio, dal numero esadecimale corrispondente alla traccia, da un secondo spazio e dal numero esadecimale corrispondente al settore. Per esempio per leggere il primo settore della traccia 18 la sintassi sarà la seguente:

R 12 01 [Return]

Come si fa a sapere qual'è il numero esadecimale corrispondente al valore decimale di una data traccia o settore? Elementare, Watson, si usa il comando cancelletto (#) che è stato inserito apposta.

Dopo aver dato il comando R, e ammesso che la sintassi sia stata rispettata, la spia rossa del drive si accenderà, il computer caricherà in memoria il contenuto di un intero settore di disco, cioè 256 byte ($00-$FF) ma sembrerà che nulla sia realmente accaduto.

Il settore caricato può ora essere visualizzato mediante il comando MEMORY DISPLAY, la cui sintassi è: M [punto iniziale] [punto finale]

Se M non è seguito da alcun valore, sullo schermo verrà visualizzato il contenuto del settore a partire da $00 fino a $FF. E' consigliabile però esaminare a spezzoni il contenuto del buffer, per esempio con M 00 60, eccetera.

Sullo schermo appariranno tante righe, ciascuna composta da:

〉 [segno di maggiore
: [segno di doppio punto
00 [valore esadecimale della posizione del primo byte del buffer visualizzato.
Otto coppie contigue dei valori esadecimali di altrettanti byte contenuti nel buffer caricato tramite READ.
Otto byte (in reverse) rappresentanti gli stessi se si tratta di caratteri stampabili.

A questo punto è possibile modificare i singoli valori, spostandosi con i tasti cursore sul valore desiderato. Ogni cambiamento dei valori visualizzati, seguito dalla pressione del tasto Return, verrà interpretato dal computer come un comando di modifica del contenuto del buffer.

Se per esempio stiamo leggendo la traccia 18 settore 1 (primo blocco della directory) e vediamo che il titolo di un programma è, per esempio, "ALGA", visualizzato in caratteri esadecimali come 41 4C 47 41, se trasformiamo il 47 in 46,il nome del programma diverrà "ALFA".

Effettuate le modifiche desiderate, se si vuole che queste abbiano un carattere permanente, bisogna riscrivere il contenuto del buffer sul disco.

A questo scopo è presente il comando WRITE, la cui sintassi è esattamente uguale a quella del comando READ. Con W 12 01 riscriveremo quindi il contenuto del buffer sul disco.

Non ci resta altro che vedere se il cambiamento è stato effettuato mediante il comando chiocciolina - dollaro che visualizza la directory.

Fintantochè non diverrete abili nell'uso del Monitor raccomandiamo di effettuare esperimenti su dischi "cavia". Dopo ogni modifica caricare i file modificati per controllare se "funzionano" ancora.

Attenzione: se sbaglierete ad indicare traccia o settore con il comando WRITE, il contenuto del disco sarà irrimediabilmente rovinato!

```
100 REM  *********************
    *********
110 REM       MONITOR ESADECIMALE
     PER
120 REM       SETTORI DI DISCO BY
     API
130 REM  *********************
140 DATA  017,008,194,007,158,0
    50,048,054,055,058,137
150 DATA  143,032,065,080,073,0
    00,000,000,169,174,224
160 DATA  160,011,032,030,171,1
    62,013,169,062,032,074
170 DATA  217,009,169,000,141,0
    01,002,032,033,010,102
180 DATA  201,062,240,249,201,0
    32,240,245,162,007,103
190 DATA  221,250,011,208,012,1
    42,000,002,189,003,014
200 DATA  012,072,189,011,012,0
    72,096,202,016,236,150
210 DATA  076,026,008,133,151,0
    32,096,009,185,019,223
220 DATA  012,032,202,009,200,2
    08,003,238,001,002,139
230 DATA  198,151,208,237,096,0
    32,101,009,169,000,177
240 DATA  133,144,169,036,141,2
    11,008,169,211,133,075
250 DATA  187,169,008,133,188,1
    69,001,133,183,169,060
260 DATA  008,133,186,169,096,1
    33,185,032,213,243,118
270 DATA  165,186,032,180,255,1
    65,185,032,150,255,069
280 DATA  164,144,208,060,160,0
    06,140,211,008,032,109
290 DATA  165,255,174,212,008,1
    41,212,008,164,144,203
300 DATA  208,042,172,211;008,1
    36,208,234,172,212,067
310 DATA  008,032,205,189,169,0
    32,032,210,255,032,140
320 DATA  165,255,166,144,208,0
    18,170,240,006,032,124
330 DATA  210,255,076,180,008,1
    69,013,032,210,255,128
340 DATA  160,004,208,198,032,0
    66,246,076,026,008,000
350 DATA  000,000,032,236,009,1
    41,014,009,164,211,048
360 DATA  177,209,201,032,208,0
    14,173,014,009,141,154
370 DATA  013,009,169,000,141,0
    14,009,076,247,008,174
380 DATA  032,236,009,141,013,0
```

```
    09,169,013,032,210,096
390 DATA  255,169,035,032,210,2
    55,174,013,009,173,045
400 DATA  014,009,032,205,189,0
    76,026,008,000,000,047
410 DATA  162,000,142,013,009,1
    42,014,009,032,207,218
420 DATA  255,201,048,144,026,2
    01,058,176,022,041,148
430 DATA  015,162,017,208,005,1
    44,002,105,009,074,229
440 DATA  110,014,009,110,013,0
    09,202,208,242,240,133
450 DATA  223,169,013,032,210,2
    55,169,036,032,210,069
460 DATA  255,173,014,009,032,2
    02,009,173,013,009,121
470 DATA  032,202,009,076,026,0
    08,032,236,009,144,006
480 DATA  003,153,019,012,200,1
    98,151,096,032,096,192
490 DATA  009,169,032,076,210,2
    55,169,013,076,210,195
500 DATA  255,160,000,140,003,0
    02,136,140,004,002,074
510 DATA  032,207,255,201,013,2
    40,023,032,236,009,224
520 DATA  144,018,141,003,002,0
    32,207,255,201,013,248
530 DATA  240,008,032,236,009,1
    44,003,141,004,002,051
540 DATA  172,003,002,032,148,0
    11,032,164,011,152,215
550 DATA  032,202,009,032,096,0
    09,169,008,032,074,151
560 DATA  008,032,101,011,076,1
    48,009,076,026,008,239
570 DATA  032,236,009,144,248,1
    68,169,008,133,151,018
580 DATA  032,033,010,032,033,0
    10,032,081,009,208,224
590 DATA  248,032,101,011,076,0
    26,008,072,074,074,210
600 DATA  074,074,032,226,009,1
    70,104,041,015,032,009
610 DATA  226,009,072,138,032,2
    10,255,104,076,210,052
620 DATA  255,024,105,246,144,0
    02,105,006,105,058,026
630 DATA  096,169,000,141,002,0
    02,032,033,010,201,174
640 DATA  032,208,009,032,033,0
    10,201,032,208,015,012
650 DATA  024,096,032,022,010,0
    10,010,010,010,141,109
660 DATA  002,002,032,033,010,0
    32,022,010,013,002,158
670 DATA  002,056,096,201,058,0
    08,041,015,040,144,149
680 DATA  002,105,008,096,032,2
    07,255,201,013,208,103
690 DATA  248,104,104,076,026,0
    08,032,207,255,201,237
700 DATA  036,208,003,076,096,0
    08,201,013,208,039,120
710 DATA  169,000,133,144,032,1
    01,009,169,008,133,130
720 DATA  186,032,180,255,169,1
    11,133,185,032,150,153
730 DATA  255,032,165,255,036,1
    44,112,005,032,210,222
740 DATA  255,208,244,032,171,2
    55,076,026,008,201,196
750 DATA  036,240,029,072,169,0
    08,133,186,032,177,058
760 DATA  255,169,111,133,185,0
    32,147,255,104,032,143
770 DATA  168,255,032,207,255,2
    01,013,208,246,032,081
780 DATA  174,255,076,026,008,0
    32,033,010,032,236,114
790 DATA  009,144,245,141,245,0
    11,032,033,010,032,134
800 DATA  236,009,144,234,141,2
    48,011,032,041,011,083
810 DATA  173,000,002,201,001,2
    40,030,169,049,032,129
820 DATA  226,010,162,013,032,1
    98,255,162,000,032,066
830 DATA  207,255,157,019,012,2
    32,208,247,032,204,037
840 DATA  255,032,078,011,076,0
    26,008,032,020,011,037
850 DATA  162,013,032,201,255,1
    62,000,189,019,012,021
860 DATA  032,210,255,232,208,2
    47,032,204,255,169,052
870 DATA  050,032,226,010,076,1
    90,010,141,238,011,216
880 DATA  162,015,173,245,011,0
```

```
     32,088,011,142,245,100
 890 DATA  011,141,246,011,173,2
     48,011,032,088,011,204
 900 DATA  142,248,011,141,249,0
     11,162,015,032,201,188
 910 DATA  255,162,000,189,237,0
     11,032,210,255,232,047
 920 DATA  224,013,208,245,076,2
     04,255,162,015,032,154
 930 DATA  201,255,162,000,189,2
     29,011,032,210,255,008
 940 DATA  232,224,008,208,245,0
     76,204,255,169,015,100
 950 DATA  168,162,008,032,186,2
     55,169,000,032,189,177
 960 DATA  255,032,192,255,169,0
     13,168,162,008,032,006
 970 DATA  186,255,169,001,162,0
     77,160,011,032,189,218
 980 DATA  255,076,192,255,035,1
     69,013,032,195,255,197
 990 DATA  169,015,076,195,255,1
     62,048,056,233,010,195
1000 DATA  144,003,232,176,249,1
     05,058,096,152,056,247
1010 DATA  233,008,168,032,096,0
     09,169,018,032,210,207
1020 DATA  255,162,008,185,019,0
     12,041,127,201,032,018
1030 DATA  176,004,169,046,208,0
     03,185,019,012,032,086
1040 DATA  210,255,169,000,133,2
     12,200,202,208,229,026
1050 DATA  169,146,076,210,255,1
     73,001,002,208,006,222
1060 DATA  204,004,002,176,001,0
     96,104,104,076,026,025
1070 DATA  008,032,101,009,169,0
     58,162,062,076,217,126
1080 DATA  009,147,013,068,073,0
     83,075,045,083,069,153
1090 DATA  067,084,079,082,032,0
     72,069,088,032,077,170
1100 DATA  079,078,073,084,079,0
     82,013,040,067,079,162
1110 DATA  077,077,065,078,068,0
     83,058,032,082,044,152
1120 DATA  087,044,077,044,064,0
     44,036,044,035,044,007
1130 DATA  088,044,058,041,013,0
     00,066,045,080,032,211
1140 DATA  049,051,032,048,085,0
     49,058,049,051,032,248
1150 DATA  048,032,000,000,032,0
     00,000,058,087,082,083
1160 DATA  077,064,088,036,035,0
     00,009,010,010,009,082
1170 DATA  010,227,008,009,174,1
     33,133,105,044,122,197
1180 DATA  212,014,000,063,031,0
     63,063,063,047,079,123
1190 FOR I= 2049 TO 3094 STE
     P 10:Z=0
1200 FOR J=0 TO 9:READ X:Z=(
     Z+X) AND 255:CK=CK+X:NEXT
      J
1210 READ W:RI=PEEK (63)+PEEK (
     64)*256:PRINT CHR$ (147);"R
     IGA: ";RI
1220 IF W<>Z THEN PRINT "ERRO
     RE RIGA";RI-10;"-";RI+10:ST
     OP
1230 NEXT I: IF CK<>107333 THE
     N PRINT "ERRORE SOMMA DATI
     ":STOP
1240 PRINT CHR$ (147)"IL CODICE
     SORGENTE NON CONTIENE ERROR
     I"
1250 PRINT "INSERIRE UN DISCO E
     PREMERE UN TASTO"
1260 POKE 198,0:WAIT 198,1:POK
     E 198,0
1270 RESTORE :OPEN 15,8,15,"I0"
1280 OPEN 8,8,8,"DISKMON,P,W"
1290 INPUT# 15,E1,E1$
1300 IF E1 THEN CLOSE 8:CLOS
     E 15:PRINT "ERRORE DISCO:"
     E1;E1$:STOP
1310 PRINT# 8,CHR$ (1);CHR$ (8);
     :REM ATTENZIONE AL PUNTO E
      VIRGOLA!
1320 FOR I= 2049 TO 3094 STE
     P 10
1330 FOR J=0 TO 9:READ X:PRI
     NT# 8,CHR$ (X);:REM ATTENZ
     IONE AL PUNTO E VIRGOLA!!
1340 NEXT J:READ W:NEXT I
1350 CLOSE 8:INPUT# 15,E1,E1$
1360 PRINT "STATO DISCO: "E1;E1$
     :CLOSE 15
```

Traccia 18 e dintorni

Una manciata di mini programmi di straordinario interesse

di Paolo Agostini

Sicuramente, leggendo l'articolo "Smascheriamo il drive" apparso su CCC N.30, alcuni di voi avranno provato a fare i primi esperimenti col floppy disk drive per conoscere meglio il suo funzionamento e sfruttare a fondo le possibilità che offre.

Questo articolo approfondisce alcuni argomenti, tra cui il funzionamento di comandi speciali che consentiranno di applicare protezioni e "trucchetti" allo scopo di conferire un aspetto particolare alle directory dei dischi.

Ponete la massima attenzione nella copiatura dei programmi perchè sono "critici" e basta un piccolo errore per causarne il malfunzionamento, il blocco o (peggio) la distruzione dei dati presenti sul floppy.

E' implicito il consiglio di utilizzare, almeno per i primi esperimenti, un disco "cavia" dal momento che l'impiego dei programmi proposti, se mal copiati o usati, può causare la perdita irreparabile dei vostri preziosi file.

1: Memorizza Directory

Cominciamo con un listatino in Basic per esaminare la directory dei dischi che può essere facilmente inserito come sub-routine in programmi di vostra creazione allo scopo di leggere e/o controllare il contenuto della directory. Il programma, "inventato" ai tempi del Commodore PET, seppur valido risente del peso degli anni, ma può venire agevolmente modificato per scrivere un programma che dia la possibilità di scegliere tra file diversi contenuti sullo stesso disco.

Il nome dei singoli file viene infatti memorizzato nell'array NF$(144) da porre nelle prime righe del programma principale.

2: Visualizza Directory

Il secondo listato, più breve, è una trasposizione in Basic del sistema che viene normalmente usato in linguaggio macchina. E' molto più veloce del precedente e non richiede array dal momento che l'unico suo compito è quello di visualizzare la directory sullo schermo. CCC ha già pubblicato più di una volta utility di questo genere ma la riportiamo egualmente per motivi di completezza.

3: Esame protezione disco

Per essere certi di accedere al disco, bisogna che questo sia privo di quell'etichetta da apporre sul foro laterale e che in inglese si chiama "write-protect tab": ha il compito di impedire di scrivere sul disco o di cancellarlo inavvertitamente. Con il miniprogramma riportato è possibile accertarsi della sua presenza senza uscire dal programma principale e senza incorrere in errori che eventualmente potrebbero riuscire fatali alla registrazione dei dati.

Formattando...

E veniamo alla formattazione del disco. Tutti coloro che hanno un floppy disk sanno di certo che per formattare un disco si impartisce il comando:

```
OPEN 15,8,15,"N0:nome disco,ID"
CLOSE 15
```

in cui "ID" è un identificatore formato da due lettere. Saprete anche, per esperienza, che non è possibile utilizzare simboli semi-grafici.

Esiste però un sistema per utilizzarli, ed è il seguente:

```
OPEN 15,8,15,"N0:nome disco"+
CHR$(34)+",ID"
CLOSE15
```

in cui, al posto dei caratteri alfabetici "ID", inserirete, magari, due cuoricini (Shift + "S").

Fate molta attenzione al punto in cui viene inserita la virgola, pena il non funzionamento del trucchetto. Con questo sistema potrete usare qualsivoglia segno grafico della tastiera del computer in modo da "personalizzare" le directory.

A proposito di formattazione, ri-

cordiamo che ve ne sono due diversi tipi: quella "lunga" e quella "corta".

La formattazione "lunga", vale a dire il famigerato "NEW:nome disco.ID" si rende necessaria soltanto quando si formatta un nuovo disco.

Quando invece si tratta di "riformattare", ovvero di cancellare un disco già formattato, è sufficiente dare il comando "NEW:nome disco" e basta. Il disk drive ronzerà per qualche istante e dopo un paio di secondi potrete leggere la directory completamente vuota, con un notevole risparmio di tempo e di logorio del drive.

4: Cambio nome disco

Il programma in oggetto è talmente semplice che non merita commenti. E' ovvio che si limita a modificare soltanto il nome del floppy: tutti i file ivi presenti rimangono rigorosamente inalterati.

5: ID con cinque caratteri

Sapevate che è possibile assegnare, come identificatore, sino ad un massimo di cinque caratteri alla "testata" della directory?

Sarà quindi possibile, con questo listatino, personalizzarla con "qualcosa" che identifichi inequivocabilmente i vostri dischi, utilizzando un ID come "Pino" o "Roby" o "Stefi" o quello che preferite.

6: Proteggi Directory

Quando si carica in memoria la directory con il comando LOAD"$",8 (e poi LIST), la stampa sullo schermo avviene con lo stesso sistema con cui vengono listati i programmi Basic.

Ciascuna riga Basic viene infatti memorizzata nel computer con uno zero finale, e la fine del programma viene indicata con altri due zeri.

Quando viene chiamata la routine di LIST, questa visualizza le singole righe, andando a capo quando incontra uno zero e la visualizzazione si interrompe del tutto quando il computer incontra i famosi tre zeri posti in successione.

Il compito che svolge il programma proposto è quello di "scrivere" tre zeri in un punto della traccia 18 del disco (dove non possiamo "disturbare" la directory) in modo che, impartendo LIST, il computer incontra i tre zeri, crede che la directory finisca lì e ne interrompe la visualizzazione.

Potrete così impedire la lettura dei file contenuti sul disco e l'unico modo che avrà il malcapitato ficcanaso di utilizzarli sarà quello di caricare alla cieca il primo file mediante un LOAD"*",8,1.

Voi, naturalmente, provvederete a mettere come primo "oggetto" del disco un file sequenziale oppure, meglio, un autoboot che impedirà ogni ulteriore accesso all'elenco dei programmi mediante il blocco del tasto RUN/STOP.

Non sapete come si fa a bloccare il tasto di RUN/STOP del C/64?Siete proprio una frana, ragazzi. Provate con POKE 808,239. E come si fa a tornare allo stato normale? Ma con POKE 808,237 perbacco!

Naturalmente se l'utilizzatore ha familiarità con i monitor e con il drive, si renderà ben presto conto di come deve fare per leggere la directory malgrado la protezione descritta.

7: Proteggendo...

A proposito di protezioni: uno dei primi tipi di protezione utilizzati fino a qualche anno fa, prima dell'avvento dei copiatori copiatutto, era il seguente.

Come avrete notato nel leggere la directory, sulla prima riga dopo il nome del disco e le due lettere che compongono l'ID vi sono altre due lettere, normalmente "A2", denominate "identificatore di formato".

Ogni computer, o meglio, ogni drive, formatta i suoi dischi in modo diverso. Il drive 1541 utilizza, appunto, il formato A2. Vale a dire che il disk drive può leggere e scrivere su dischi che utilizzino questo formato.

Il disk drive 1541 ha anche la possibilità di leggere dischi formattati con altri disk drive, come il 4040 della Commodore, ma non può scrivere su questi dischi.

Se, infatti, si tenta comunque di scrivere su dischi dotati di identificatore di formato diverso da A2, si ottiene il segnale di errore 73,DOS MISMATCH,18,00.

I primi copiatori apparsi sul mercato cominciavano sempre a copiare a partire dalla traccia 18, trascurando il fatto che il disk drive, prima di scrivere qualsiasi "cosa" su disco, effettua SEMPRE un controllo sul suo formato.

I primi copiatori, quindi, copiavano la traccia 18, ma quando tentavano di trascrivere la traccia successiva, il disk drive si bloccava irrimediabilmente segnalando l'errore.

Il sistema descritto non ha più alcun valore come protezione poichè i moderni copiatori apparsi di recente sul mercato sono oramai sofisticatissimi. Può, comunque, rivelarsi molto utile per evitare di scrivere qualcosa su un disco che vogliamo proteggere da sovrascritture.

Il programma pubblicato serve, appunto, a trasformare il formato A2 in un formato diverso, e funziona come un blocco contro la scrittura. Per "sbloccare" il disco, e renderlo nuovamente accessibile alla scrittura di nuovi file o alla cancellazione di quelli contenuti, bisogna ricorrere ad un artifizio particolare: "raggirare" il disk drive e, col comando M-W (memory-write), inserire nella sua memoria il formato "A2", facendogli credere di avergli "dato in pasto" un dischetto del tutto normale.

A questo punto diventa facile riportare l'indicatore di formato allo stato originale, e il gioco è fatto.

Una protezione che questo sistema può garantirvi consiste nel fatto che, anche se con un copiatore sarà sempre possibile duplicare il vostro dischetto, quasi nessuno sarà in grado di cancellare il vostro nome dallo stesso disco. E' infatti impossibile accedere alla scrittura sul disco (e sulle copie che ne verranno fatte!).

Inoltre il disco rimane apparentemente del tutto normale, dato che l'indicatore di formato A2 continua ad apparire regolarmente sulla testata (header) della directory. Il che, per un sistema che utilizza poco più di una decina di righe di Basic, non è poco.

```
   10 REM   *** LETTURA DIRECTORY
        ***
   20 DIM NF$(144),FT$(144):REM
        ARRAYS PER IL NOME DEI FILE
        S
   30 SP$="":FOR I=1 TO 16:SP$=SP
        $+CHR$(32):NEXT
   50 REM  QUI VA IL VOSTRO PROGR
        AMMA
   60 GOSUB  30030:REM  LETTURA D
        ISCO
   70 REM  QUANDO RITORNEA' DALLA
         ROUTINE
   80 REM  NT CONTERRA' IL NUMERO
         TOTALE DEI FILES LETTI
   90 REM  NF$(DA 1 A NT) IL NOME
         DEI FILES
  100 REM  FT$(DA 1 A NT) I FORMA
        TI DEI FILES (PRG/SEQ/REL/U
        SR/DEL)
  110 END
  120 :
30000 :
30010 REM  LETTURA NOME DEL DISCO
30020 :
30030 OPEN 1,8,0,"$"
30040 FOR I=1 TO 33
30050 GET #1,W$:ND$=ND$+W$
30060 NEXT:CLOSE 1
30070 PRINT"[CLEAR]NOME DEL DISCO
        :"ND$
30080 :
30090 :
30100 REM  LETTURA NOME FILES DEL
         DISCO
30110 :
30120 OPEN 1,8,0,"$"
30130 GET #1,W$:GET #1,W$:GOSUB 3
        0230
30140 GOSUB 30230:IF FL=1 THEN 30
        180
30150 NT=NT+1:NF$(NT)=NF$:FT$(M)=
        FT$
30160 PRINTNF$;LEFT$(SP$,16-LEN(N
        F$));FT$
30170 GOTO 30140
30180 CLOSE 1
30190 RETURN
30200 :
30210 REM  ROUTINE PER LETTURA NO
        MI FILES
30220 :
30230 NF$="":FOR I=1 TO 4:GET #1,
        W$:NEXT
30240 GET #1,W$:IF W$="" THEN FL=
        1:RETURN
30250 IF W$<>CHR$(34) THEN 30240
30260 GET #1,W$
30270 IF W$=CHR$(34) THEN 30290
30280 NF$=NF$+W$:GOTO 30260
30290 GET #1,W$:IF W$=CHR$(32) TH
        EN 30290
30300 FT$=W$
30310 GET #1,W$:FT$=FT$+W$:GET #1
        ,W$:FT$=FT$+W$
30320 GET #1,W$:IF W$<>"" THEN 30
        320
30330 RETURN
```

```
   10 REM  QUI VA IL VOSTRO PROGR
        AMMA
   20 GOSUB  30000
   30 END
   40 :
30000 REM  SUBROUTINE LETTURA DIR
        ECTORY
30010 REM  (SENZA UTILIZZO MEMORI
        A)
30020 OPEN 1,8,0,"$":GET #1,W$:GE
        T #1,W$
30030 GET #1,Y$:GET #1,W$:PRINTVA
        L(Y$);
30040 FOR X=0 TO 27:GET #1,W$:PRI
        NTW$;:NEXT:PRINT
30050 IF ST=64 THEN 30130
30060 GET #1,W$:GET #1,W$
30070 GET #1,LO$:GET #1,HI$
30080 LO$=LO$+CHR$(0):HI$=HI$+CHR
        $(0)
30090 PRINTASC(LO$)+256*ASC(HI$);
30100 FOR X=0 TO 27:GET #1,W$
30110 PRINTW$;:NEXT:PRINT
30120 GOTO 30050
30130 CLOSE 1
30140 RETURN
```

```
  100 REM  CONTROLLO SE IL DISCO
        E' PROTETTO DALL'ETICHETTA
        CONTRO LA SCRITTURA
  110 REM  (WRITE-PROTECT TAB ON)
  120 REM  SE L'ERRORE E' IL 26,
```

computer service

VENDITA PER CORRISPONDENZA

ACCESSORI PER COMPUTER COMMODORE

GRUPPO CONTINUITÀ
Fornito senza le 12 batterie a stilo ricaricabili. Consente il funzionamento del Vostro computer Commodore C64 o VIC 20 in assenza di corrente. Durata di funzionamento 30 minuti. Ricarica tramite alimentatore Commodore.

KIT ALLINEAMENTO TESTINA
Composto dal cacciavite, nastro di controllo e strumento di taratura con monitor audio permette il perfetto allineamento dei registratori digitali anche con nastri commerciali.

VELOCIZZATORE DI CARICAMENTO FLOPPY
Cartridge con un insieme di utility residenti su ros per velocizzare il drive nel Commodore 64.

INTERFACCIA RADIO
Indispensabile per registrare con registratore Commodore modello "C2N" i programmi speciali per computer trasmessi dalle emittenti radio.

CUFFIA PER COMMODORE C 64
Leggerissima permette l'ascolto personale del computer evitando di disturbare durante i giochi.

COPIATORE PROGRAMMI
Dispositivo hardware per effettuare copie di nastri protetti o turbo utilizzando due registratori Commodore o compatibili.

DUPLICATORE CASSETTE
Indispensabile per realizzare delle copie, con un registratore normale, di un nastro protetto o con caricamento turbo.

Articolo	Art.	L.
Bus quadrislot	Art. CD 100	L. 55.000
Interfaccia cassette	Art. CD 101	L. 30.000
Duplicatore cassette	**Art. CD 102**	L. 30.000
Copiatore programmi	**Art. CD 103**	L. 30.000
Interfaccia radio	**Art. CD 104**	L. 30.000
Kit allineamento testina	**Art. CD 105**	L. 47.000
Alimentatore per C64 e VIC 20	Art. CD 106	L. 45.000
Gruppo continuità (fornito senza le 12 batterie a stilo ricaricabili)	**Art. CD 107**	L. 66.000
Pacco batterie (12 stilo 1,2 Volt ricaricabili	Art. CD 117	L. 52.000
Commutatore antenna TV/computer	Art. CD 108	L. 9.500
Tasto reset	Art. CD 109	L. 5.500
Interfaccia Centronics	Art. CD 112	L. 104.000
Espansione di memoria per C 16	Art. CD 114	L. 158.000
Velocizzatore di caricamento flop.	**Art. CD 115**	L. 49.000
Espansione di memoria per VIC 20 16K	Art. CD 116	L. 112.000
Modulatore Executive	Art. CD 120	L. 72.000
Penna ottica grafica	Art. CD 121	L. 45.000
Tavoletta grafica	Art. CD 130	L. 238.000
Multipresa con filtro - 2 prese	Art. CD 140	L. 41.000
Cuffia per Commodore C 64	**Art. CD 150**	L. 19.000
Stabilizzatore elettronico di tensione 500 W	Art. CD 160	L. 430.000
Gruppo di cintinuità 60 W	Art. CD 170	L. 400.000
Gruppo di continuità 200 V	Art. CD 180	L. 802.000
Inventer 12 Volt cc. 220 Volt ca. 100 Watt	Art. CD 190	L. 297.000
Cavo alimentazione	Art. CD 200	L. 4.600
Cavo drive o stampante Commodore	Art. CD 205	L. 8.500
Prolunga per Joystick - mt. 3	Art. CD 210	L. 25.000
Prolunga per cavo TV - mt. 3	Art. CD 215	L. 12.500
Cavo audio - mt. 6	Art. CD 220	L. 15.500
Adattatore Joistick (Atari e C64 al C 16)	Art. CD 225	L. 10.500
Adattatore registratore per C 16	Art. CD 226	L. 19.500
Mascherina antirifiesso 12"	Art. CD 300	L. 35.000
Nastro inchiostrato per Tally - mt. 80	Art. CD 610	L. 16.500
Nastro inchiostrato per Tally - mt. 180	Art. CD 611	L. 16.500
Nastro inchiostrato per Tally 1000 e Honeywell	Art. CD 612	L. 9.500
Nastro inchiostrato per Commodore MRS 801	Art. CD 614	L. 13.000
Nastro inchiostrato per Commodore MPS 802	Art. CD 616	L. 18.000
Nastro inchiostrato per Commodore MPS 803	Art. CD 618	L. 19.500
Mause per Commodore C 64	Art. CD 860	L. 240.000
Pacco carta lettura facilitata 24" x 11" modulo da 500 fogli con bordi a strappo	Art. CD 630	L. 13.500
Supporto stampante porta carta in plexiglass "fume" - normale	Art. CD 660	L. 59.000
Supporto stampante porta carta in plexiglass "fume" - rinforzato	Art. CD 670	L. 80.000
Floppy disk 5" singola faccia doppia densità "ODP" - conf. 10 pezzi	Art. CD 700	L. 40.000
Floppy disk 5" singola faccia doppia densità "CBS" - conf. 10 pezzi	Art. CD 702	L. 38.000
Floppy disk 5" singola faccia doppia densità "VERBATIM" - conf. 10 pezzi	Art. CD 704	L. 42.000
Floppy disk 5" singola faccia doppia densità "DYSAN" - conf. 10 pezzi	Art. CD 706	L. 68.000
Nastri magnetici C 10 digitali - conf. 10 pezzi	Art. CD 712	L. 20.000
Nastri magnetici C 15 digitali	Art. CD 714	L. 21.000
Copritastiera in plexiglass per C64 - C16 e VIC 20	Art. CD 750	L. 16.000
Copritastiera in stoffa per C64 - C16 e VIC 20	Art. CD 760	L. 10.500
Vaschetta portafloppy in plexiglass per 40 dischi con chiave	Art. CD 770	L. 30.000
Vaschetta portafloppy in plexiglass per 90 dischi con chiave	Art. CD 780	L. 37.000
Kit pulizia testine registratore	Art. CD 815	L. 13.500
Kit pulizia disk drive	Art. CD 820	L. 26.000
Kit pulizia tastiera	Art. CD 830	L. 16.500
Foratore disk in plastica (per utilizzare la seconda faccia dei dischi)	Art. CD 840	L. 10.000
Foratore disk in metallo "tako"	Art. CD 849	L. 14.000
Joystick Spectravideo II	Art. CD 850	L. 27.000
Joystick a Microswitch	Art. CD 851	L. 52.500
Joystick senza fili con unità ricevente (funziona a batteria)	Art. CD 852	L. 98.000
Joystick per Commodore 16 (originale)	Art. CD 130	L. 29.500

TUTTI I PREZZI SONO COMPRENSIVI DI IVA
NON SI ACCETTANO ORDINI INFERIORI A L. 30.000
CONTRIBUTO FISSO SPESE DI SPEDIZIONE L. 5000

SI ACCETTANO ANCHE ORDINI TELEFONICI AI NUMERI 0522/661647-661471

BUONO DI ORDINAZIONE

NOME - COGNOME

INDIRIZZO

N

C.A.P.

CITTA

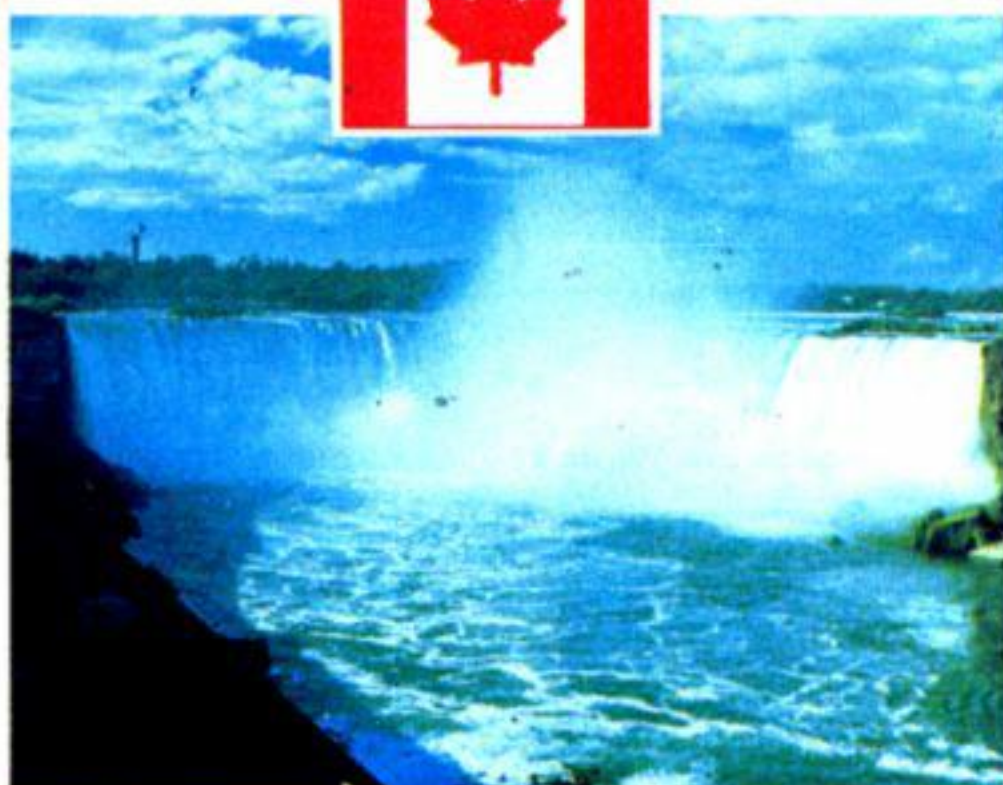
PROVINCIA

VOGLIATE INVIARMI IN CONTRASSEGNO

N.	Art.	L.
N.	Art.	L.
N.	Art.	L.
SPESE SPEDIZIONE		L. 5.000
PAGHERÒ AL POSTINO		L.

COMPUTER SERVICE VIA A. MANZONI, 49 - 42017 NOVELLARA (RE) - TEL. (0522) 661647

SUL SENTIERO DELLE GIUBBE ROSSE

Una vera esperienza di vita per i ragazzi/e oltre i 10 anni
Abbinare lo studio della Lingua Inglese, al contatto di una natura incontaminata
Una vacanza-studio unica ed indimenticabile, in uno scenario che non ha confronti.

Questo tipo di vacanza è indirizzato sia ai principianti, sia a coloro che hanno già maturato una conoscenza della Lingua Inglese, ma il denominatore comune è il reale contatto con la natura.

- Un viaggio di 19 giorni attraverso la Provincia dell'Ontario a bordo di un "Super Van" da 15 posti, con aria condizionata e stereo system, con l'assistenza di personale qualificato.

Ogni 10 partecipanti ci sono 4 persone di assistenza.
Le attività standard includono:

- partecipazione alla vita di campeggio, canoa, tracking, pesca, white-water rafting, ginnastica, nuoto e studio della Lingua Inglese.

Quest'ultimo aspetto sarà curato da insegnanti specializzati dello SHENKER INSTITUTE OF ENGLISH, con due ore di lezione al giorno, seguendo il METODO SANDWICH di GEORGE SHENKER.

- Viaggio Andata/Ritorno MILANO/TORONTO in classe turistica.
- Tre pasti al giorno dalla 1ª colazione del secondo giorno, al lunch del 17 giorno.*
- Full-Day Immersion di Lingua Inglese con personale SHENKER più 2 ore al giorno di corso intensivo.
- Materiale audio-didattico per il Corso comprendente:
  * Walkmann
  * Cassette
  * Libri di testo e work book
- Assicurazione completa EUROPE-ASSISTANCE.
- Pernottamenti in hotel di categoria superiore e in Campeggi nei "NATIONAL PARKS".
- Tutte le tasse d'iscrizione. attrezzatura da pesca
  * canna da pesca con mulinello
  * licenza di pesca

I costi di noleggio del Super Van e tutti i costi annessi:
* benzina
* autostrade con pedaggio

Il programma si divide in 4 diversi turni a partire dall'ultima settimana di giugno
Prenotazioni e informazioni presso:

SHENKER INSTITUTE OF ENGLISH - Corso Monforte, 36 (MI) - Tel. 02/700332/700363/700929
ore ufficio - Sig.ra Sawchik - Olivieri (ore serali) Tel. 039/513211

UVET - Viale Ferdinando di Savoia, 4 (MI) - Tel. 02/675061 (30 linee)
ore ufficio - Sig. Biagi

SYSTEMS - Viale Famagosta, 75 (MI) - Tel. 02/8467348/9
ore ufficio - Sig. Tidone

in collaborazione con:
SHENKER INST. OF ENGLISH
CP AIR
UVET

```
    ALLORA L'ETICHETTA C'E'!
130 :
140 OPEN 15,8,15,"I0":OPEN 8,8,
    8,"#"
150 PRINT#15,"U1 8 0 18 0"
160 PRINT#15,"B-P 8 0"
170 PRINT#15,"U2 8 0 18 0"
180 INPUT#15,E1,E1$:CLOSE 8:CLO
    SE 15
190 :
200 RM$="RIMUOVERE L'ETICHETTA
    PER CONSENTIRE    L'ACCESSO
     AL DISCO"
210 IF  E1=26  THEN  PRINT RM$
```

```
100 REM  CAMBIO NOME DISCO
110 :
120 REM  LEGGE IL NOME DEL DISC
    O
130 OPEN 15,8,15,"I0"
140 OPEN 8,8,8,"#"
150 PRINT#15,"U1 8 0 18 0"
160 PRINT#15,"B-P 8 144"
170 DN$="":FOR Y=1 TO 16:X$=""
180 GET #8,X$:IF X$=CHR$(160) T
    HEN 200
190 DN$=DN$+X$
200 NEXTY
210 :
220 :
230 REM  CHIEDE NUOVO NOME E LO
     FORMATTA
240 PRINT"[CLEAR][2 DOWN]NOME A
    TTUALE DEL DISCO: ";DN$
250 INPUT "[2 DOWN]NUOVO NOME D
    EL DISCO: ";NN$
260 IF NN$="" THEN NN$=DN$
270 IF LEN(NN$)<16 THEN NN$=NN$
    +CHR$(160):GOTO 270
280 IF LEN(DN$)>16 THEN NN$=LEF
    T$(NN$,16)
290 :
300 :
310 REM  SCRIVE NUOVO NOME SUL
    DISCO
320 PRINT#15,"B-P 8 144"
330 PRINT#8,NN$;:REM  PUNTO E V
    IRGOLA!
340 PRINT#15,"B-P 8 0"
350 PRINT#15,"U2 8 0 18 0"
360 CLOSE 8
370 :
380 :
390 REM  CONTROLLO ERRORI
400 INPUT#15,E1,E1$,E2,E3
410 PRINT"[2 DOWN]";E1;",";E1$;
    ",";E2;",";E3
420 PRINT#15,"I0":CLOSE 15
430 PRINT"[2 DOWN]ORA IL DISCO
    SI CHIAMA: ";NN$
440 END
```

```
100 REM  CAMBIO ID DEL DISCO
110 :
120 REM  LEGGE LA VECCHIA ID
130 OPEN 15,8,15,"I0"
140 OPEN 8,8,8,"#"
150 PRINT#15,"U1 8 0 18 0"
160 PRINT#15,"B-P 8 162"
170 GET #8,I1$:IF I1$="" THEN I
    1$=CHR$(0)
180 GET #8,I2$:IF I2$="" THEN I
    2$=CHR$(0)
190 :
200 :
210 REM  CHIEDE NUOVA ID E LA F
    ORMATTA
220 PRINT"[CLEAR][2 DOWN]ID ATT
    UALE DEL DISCO: ";I1$;I2$
230 INPUT "[2 DOWN]NUOVA ID DEL
     DISCO: ";ID$
240 IF ID$="" THEN ID$=I1$+I2$
250 IF LEN(ID$)>5 THEN NN$=LEFT
    $(ID$,5)
260 :
270 :
280 REM  SCRIVE NUOVA ID SUL DI
    SCO
290 PRINT#15,"B-P 8 162"
300 PRINT#8,ID$;:REM  PUNTO E V
    IRGOLA!
310 PRINT#15,"B-P 8 0"
320 PRINT#15,"U2 8 0 18 0"
330 CLOSE 8
340 :
350 :
360 REM  CONTROLLO ERRORI
370 INPUT#15,E1,E1$,E2,E3
380 PRINT"[2 DOWN]";E1;",";E1$;
    ",";E2;",";E3
```

```
390 PRINT#15,"I0":CLOSE 15
400 PRINT"[2 DOWN]LA NUOVA ID O
    RA E'  : ";ID$
410 END
```

```
100 REM  "CANCELLAZIONE" DIRECT
    ORY
110 :
120 PRINT"[CLEAR][2 DOWN] CANCE
    LLAZIONE/RIPRISTINO DIRECTO
    RY"
130 :
140 REM  LETTURA FORMATO DISCO
150 OPEN 15,8,15,"I0"
160 OPEN 8,8,8,"#"
170 PRINT#15,"U1 8 0 18 0"
180 PRINT#15,"B-P 8 165"
190 GET #8,I1$:IF I1$="" THEN I
    1$=CHR$(0)
200 GET #8,I2$:IF I2$="" THEN I
    2$=CHR$(0)
210 IF I1$=CHR$(0) AND I2$=CHR$
    (0) THEN GOTO 320
220 :
230 REM  CAMBIAMENTO FORMATO
240 PRINT#15,"B-P 8 165"
250 PRINT#8,CHR$(0);CHR$(0);:RE
    M  PUNTO E VIRGOLA!
260 PRINT#15,"B-P 8 0"
270 PRINT#15,"U2 8 0 18 0"
280 CLOSE 8:GOSUB 400
290 PRINT"[2 DOWN]LA DIRECTORY
    NON ESISTE PIU'!":END
300 :
310 REM  RIPRISTINO FORMATO
320 PRINT#15,"B-P 8 165"
330 PRINT#8,"A2";:REM  PUNTO E
    VIRGOLA!
340 PRINT#15,"B-P 8 0"
350 PRINT#15,"U2 8 0 18 0"
360 CLOSE 8:GOSUB 400
370 PRINT"[2 DOWN]LA DIRECTORY
    E' STATA RIPRISTINATA!":END

380 :
390 REM  CONTROLLO CANALE DI ER
    RORE
400 INPUT#15,E1,E1$,E2,E3
410 PRINT"[2 DOWN]";E1;",";E1$;
    ",";E2;",";E3
420 PRINT#15,"I0":CLOSE 15:RETU
    RN
```

```
100 REM  PROTEZIONE CONTRO LA S
    CRITTURA
110 PRINT"[CLEAR]PROTEZIONE/SPR
    OTEZIONE DALLA SCRITTURA"
120 :
130 REM  LEGGE IL FORMATO
140 OPEN 15,8,15,"I0"
150 OPEN 8,8,8,"#"
160 PRINT#15,"U1 8 0 18 0"
170 PRINT#15,"B-P 8 2"
180 GET #8,A$:IF A$="" THEN A$=
    CHR$(0)
190 IF A$<>"A" THEN 310
200 :
210 REM  CAMBIA IL FORMATO IN "
    E2"
220 PRINT#15,"B-P 8 2"
230 PRINT#8,"E2";:REM  PUNTO E
    VIRGOLA!
240 PRINT#15,"B-P 8 0"
250 PRINT#15,"U2 8 0 18 0"
260 CLOSE 8:GOSUB 410
270 PRINT"[DOWN]ORA SI PUO' SOL
    TANTO LEGGERE IL DISCO!"
280 END
290 :
300 REM  RIPRISTINA IL FORMATO
    "A2"
310 PRINT#15,"B-P 8 2"
320 PRINT#8,"A2";:REM  PUNTO E
    VIRGOLA!
330 PRINT#15,"B-P 8 0"
340 PRINT#15,"M-W";CHR$(1);CHR$
    (1);CHR$(1);CHR$(65)
350 PRINT#15,"U2 8 0 18 0"
360 CLOSE 8:GOSUB 410
370 PRINT"[DOWN]ORA SI PUO' UTI
    LIZZARE NORMALMENTE IL   DI
    SCO!"
380 END
390 :
400 REM  CONTROLLO CANALE DI ER
    RORE
410 INPUT#15,E1,E1$,E2,E3
420 PRINT"[2 DOWN]";E1;",";E1$;
    ",";E2;",";E3
430 PRINT#15,"I0":CLOSE 15
440 RETURN
```

File recuperato mezzo salvato

Un programma che provvede a far tornare il sorriso sulle labbra ai più distratti

di Paolo Agostini

Chissà quante volte vi sarà capitato: un attimo di disattenzione, un titolo dimenticato, un dito che sbaglia a battere il tasto, un controllo mancato ed ecco che il vostro prezioso file su disco viene annullato, distrutto, annichilito. A questo punto non c'è più nulla da fare: il comando SCRATCH dà l'addio definitivo al file. Che fare? Tutto è perduto?

No: sciogliete il nodo scorsoio che avevate stretto attorno al collo, strappate la lettera che conteneva l'addio struggente ai familiari e leggete questo articolo.

Come ragiona il disk drive

Per prima cosa sarà opportuno vedere come il disk drive esegue il comando SCRATCH, per comprendere cosa fare in casi del genere. Nella directory del disco vengono posti, ad intervalli regolari di 32 byte, i nomi dei vari file. Più esattamente il nome di ogni file è preceduto da un byte che indica il tipo di file di cui si tratta, e da due byte che indicano a quale traccia e settore si trova il primo blocco di dati che compongono il file stesso.

Esaminiamo più da vicino il byte che indica il TIPO di file. Come tutti (?) certamente saprete, un byte è composto da 8 bit.

Nel byte indicante il tipo di file ogni singolo bit ha una sua funzione, e più esattamente:

- Bit numerati da 0 a 3 : indicano il tipo di file, e cioè:

%0000=DEL
%0001=SEQ
%0010=PRG
%0011=USR
%0100=REL

Ricordiamo che in lunguaggio tecnico, i numeri preceduti dal segno % stanno ad indicare numeri binari. In questo caso, visto che il numero binario viene indicato con sole quattro

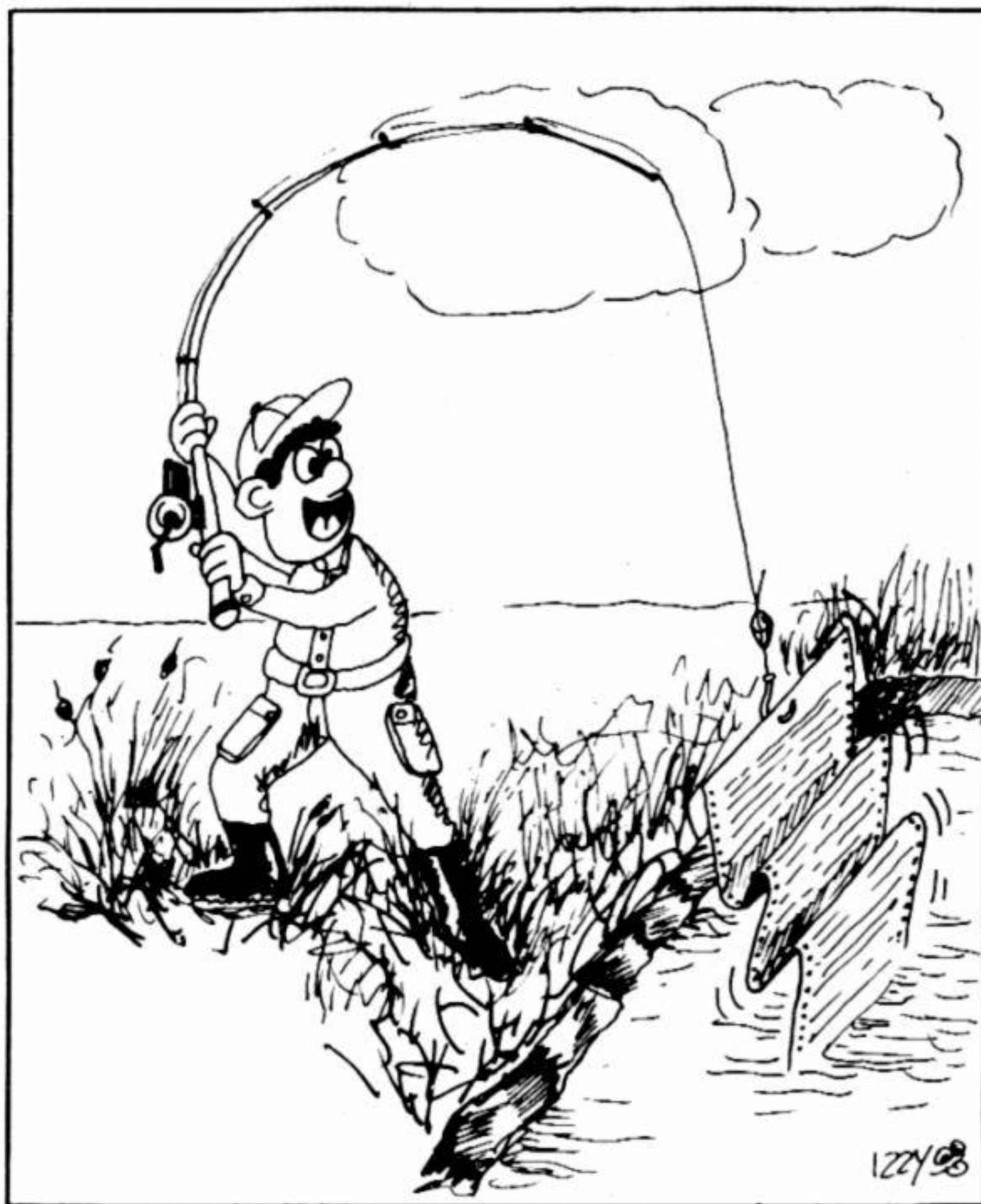
cifre, si parla di NIBBLE, cioè di mezzo byte. I corrispondenti numeri decimali e il relativo significato sono:
0=file cancellato
1=file sequenziale
2=file tipo programma
3=file user
4=file relativo
Si noti inoltre (e vedremo poi l'importanza di quanto stiamo per dire) che l'indicazione di file cancellato (DEL) impedisce di sapere la tipologia cui apparteneva il file PRIMA della cancellazione.

• bit 4 e 5 : sono bit "illegali" e venivano talvolta utilizzati in passato per la protezione di file.

• bit 6: ha la funzione di segnalare se un file è protetto contro lo SCRATCH. Se questo bit è "settato", il file non può più venire cancellato dal comando SCRATCH. In questo caso sulla Directory appare, vicino al tipo di programma, anche il simbolo "<".

• bit 7: Se il bit è "settato" il file è stato regolarmente chiuso. Se invece il file è rimasto per una qualche ragione "aperto", vale a dire se la B.A.M. (Block Availability Map, o Mappa della Disponibilità dei Blocchi) non è stata aggiornata, il file viene contrassegnato col simbolo "*" vicino al tipo di programma, come forse vi sarà capitato spesso di vedere. Tale evenienza capita se aprite un file e dimenticate di chiuderlo oppure se, mentre registrate un programma, aprite lo sportello del drive, e in altre circostanze quasi sempre legate a condizioni di errore.

Vediamo qualche esempio tra i più frequenti:

%10000010 (hex: $82, dec. #130): file di tipo "PRG" regolarmente chiuso
%11000010 (hex: $C2, dec. #194): file di tipo "PRG<" protetto
%00000010 (hex: $02, dec. #002): file di tipo "PRG*", "aperto"
%10000001 (hex: $81, dec. #129): file di tipo "SEQ" regolare
%00000000 (hex: $00, dec. #000): file cancellato
%10000000 (hex: $80, dec. #128): file illegale. Benchè si tratti di un file cancellato, appare comunque nella directory con l'indicazione "DEL" ("deleted"), ma non può venire caricato
%11000000 (hex: $C0, dec. #192): file illegale protetto di tipo "DEL<"

A questo punto avrete capito che il DOS (acronimo di Disk Operating System, o Sistema Operativo del Disco) per cancellare il file non fa altro che cambiare il bit che indica il tipo di file in uno zero. Se salviamo su disco un nuovo file, il DOS controllerà quel particolare byte, e se questo è posto a zero, il DOS scriverà il nuovo file sopra quello vecchio. In questo caso è impossibile "recuperare" il file precedentemente cancellato.

Ma se su quel disco non abbiamo scritto null'altro, il sistema per "resuscitare" il file cancellato esiste senz'altro: basterà riportare quel famoso byte al valore originale e il file cancellato riapparirà come per incanto sulla directory. Però a questo punto il salvataggio è completo solamente a metà: per evitare problemi bisogna dare il comando "VALIDATE", che aggiornerà la B.A.M. (Block Allocation Map, o Mappa dei blocchi disponibili), indicando come "occupati" i blocchi in cui si trova il nostro file redivivo.

Il programma che effettua questo "salvataggio" può essere scritto in due modi. Il primo è quello che io definisco "Io Tarzan - Tu Jane", vale a dire il programma chiede il titolo del

file da recuperare, lo cerca nella directory e lo recupera.

Il perchè lo ritenga un po' primitivo è presto spiegato: in genere si cancellano file preziosi quando si è dimenticato il nome del file. Quindi un programma che chiede il nome del file è un controsenso e generalmente porta alla conclusione logica del "FILE NOT FOUND" (Io file non trovato, come direbbe Tarzan!).

Per questa ragione ho scelto di scrivere il programma nel secondo modo possibile, ovverossia effettuando uno screening (visualizzazione) di tutti i file presenti nella directory passandola tutta in rassegna, e si decide file per file se salvarlo o meno.

E' stata inserita nel programma la possibilità di "proteggere" i file dall'azione dello SCRATCH, in maniera che quelli più preziosi non verranno più cancellati in futuro. Per cancellarli, una volta protetti, vi restano solo due modi: o usare nuovamente il programma pubblicato oppure riformattare il dischetto...

Il programma

Copiate con la massima cura il listato di queste pagine e, preparato un dischetto contenente file poco importanti, inseritelo nel drive e date il RUN.

Sullo schermo appariranno, uno alla volta, i nomi dei file presenti che siano "visibili" alla directory oppure no. Fate attenzione alla scritta DEL, quelli sono i file cancellati che potrete eventualmente recuperare.

Nel caso non vogliate modificare lo stato dei file, premete la barra spaziatrice, il tasto Return o qualunque altro tasto, per proseguire col file successivo. Per modificare lo stato dei file, premete i tasti "1", "2", "3" oppure "4":

1 Recupera i file di tipo DEL

2 Cancella QUALUNQUE tipo di file, anche quelli che è impossibile cancellare in altro modo, come i file che hanno come nome una virgola o le virgolette (esistono, esistono, e se seguirete Commodore Computer Club imparerete anche voi a registrare file dotandoli di nomi strani.

3 Protegge ogni tipo di file da cancellazioni involontarie (ma anche volontarie).

4 Consentirà di sproteggere i file protetti nel modo 3.

Noterete che sullo schermo appare (oltre al nome e tipo di file e al numero di blocchi che occupa sul disco) anche un provvidenziale "T/S".

Si tratta della TRACCIA e del SETTORE dove il file inizia, e il dato può esservi di aiuto per esaminare il disco in dettaglio mediante altri programmi di utility.

Conclusioni

Il programma presentato deve esser considerato una notevole utility da tenere sempre a portata di mano. Digitatelo con gran cura e provatelo su un disco "cavia" da sottoporre a tutte le torture cancellando, recuperando, proteggendo e sproteggendo i singoli file.

Non abbreviate il comando PRINT# con il punto di domanda (?#) per evitare Syntax Error.

Desiderando recuperare un file cancellato è indispensabile comunicare al drive il tipo di file. Come abbiamo detto all'inizio, infatti, non rimane traccia di tale informazione dopo il comando di Scratch. Se non ricordate il tipo di file provate, in più fasi, ad assegnare i 4 tipi, uno alla volta. "Usciti" dal programma potete controllare se il file era un programma Basic oppure un file sequenziale di un Word Processor oppure un file relativo di un Data Base.

Se, nonostante tutti i tentativi, non riuscite a capire che diavolo fosse, riprendete la corda messa da parte all'inizio dell'articolo e strofinate sopra, prima dell'uso, molto, molto sapone...

```
 10 REM DISK MANAGER
 20 :
100 PRINTCHR$(147):GOTO 150
110 :
120 GET #2,A$:IF A$="" THEN A$=
    CHR$(0)
130 RETURN
140 :
150 FOR I=1 TO 39:SP$=SP$+CHR$(
    32):NEXT
160 PRINTCHR$(147);SPC(2);"DISK
     FILE UTILITY BY PAOLO AGOS
    TINI"
170 FOR I=1 TO 40:PRINTCHR$(192
    );:NEXT
180 PRINT"1: RECUPERA.       2: C
    ANCELLA"
190 PRINT"3: PROTEGGI.       4: S
    PROTEGGI":PRINT"SPAZIO: NES
    SUNA MODIFICA"
200 FOR I=1 TO 40:PRINTCHR$(192
    );:NEXT
210 PRINT" FILE NAME"+LEFT$(SP$
    ,9)+"TYPE"+LEFT$(SP$,2)+"BL
    OCKS";
220 PRINTLEFT$(SP$,4)+"T/S"
230 FOR I=1 TO 40:PRINTCHR$(192
    );:NEXT
240 PRINTCHR$(19):FOR I=1 TO 14
    :PRINTCHR$(17);:NEXT
250 FOR I=1 TO 8:PP$=PP$+CHR$(1
    7):NEXT
260 P$=CHR$(19):FOR I=1 TO 8:P$
    =P$+CHR$(17):NEXT:P$=P$+CHR
    $(32)
270 :
```

```
280 REM  DIRECTORY AMPLIATA
290 OPEN 15,8,15,"I0":OPEN 2,8,
    2,"#":S=1
300 PRINT#15,"U1 2 0 18 ";S
310 PRINT#15,"B-P 2 0"
320 GOSUB 120:T=ASC(A$)
330 GOSUB 120:S1=ASC(A$)
340 FOR I=0 TO 7
350 PRINT#15,"B-P 2 ";2+32*I
360 GOSUB 120:FT=ASC(A$)
370 TY=FT:FT=(FT) AND 15
380 IF FT=0 THEN FT$="DEL"
390 IF FT=1 THEN FT$="SEQ"
400 IF FT=2 THEN FT$="PRG"
410 IF FT=3 THEN FT$="USR"
420 IF FT=4 THEN FT$="REL"
430 IF (TY AND 128)=0  AND  FT<
    >0  THEN  FT$=FT$+"*"
440 IF  TY  AND  64  THEN FT$=F
    T$+"<"
450 PRINT#15,"B-P 2 ";3+32*I
460 GOSUB 120:IT=ASC(A$):IF IT=
    0 THEN 590
470 GOSUB 120:IS=ASC(A$)
480 IT$=STR$(IT):IT$=RIGHT$(IT$
    ,LEN(IT$)-1):IT$=RIGHT$("00
    "+IT$,2)
490 IS$=STR$(IS):IS$=RIGHT$(IS$
    ,LEN(IS$)-1):IS$=RIGHT$("00
    "+IS$,2)
500 NF$="":FOR J=1 TO 16:GOSUB
    120
510 IF (A$<>CHR$(0)) AND (A$<>C
    HR$(160)) THEN  NF$=NF$+A$
520 NEXTJ
530 PRINT#15,"B-P 2 ";30+32*I
540 GOSUB 120:BT=ASC(A$)
550 GOSUB 120:BT=ASC(A$)*256+BT
560 PRINTP$;SP$;P$;LEFT$(NF$+SP
    $,18);LEFT$(FT$+SP$,7);
570 PRINTLEFT$(STR$(BT)+SP$,8);
    IT$;":";IS$
580 GOSUB 640
590 NEXTI
600 IF T=18 AND S1<21 THEN S=S1
    :GOTO 300
610 CLOSE 2:PRINTP$;SP$;P$;"ATT
    ENDERE PREGO - CONVALIDA DI
    SCO"
620 PRINT#15,"V0":CLOSE 15:END
630 REM  ROUTINE DI RECUPERO FI
    LE
640 PRINTP$;PP$;SP$;P$;PP$;"PRE
    MERE UN TASTO"
650 GET A$:IF A$="" THEN 650
660 PRINTP$;PP$;SP$
670 IF A$<>"1" THEN 790
680 PRINTP$;PP$;"RECUPERO FILE
    CANCELLATO"
690 PRINTP$;PP$;CHR$(17);"TIPO
    DI FILE: SEQ / PRG / USR /
    REL "
700 GET R$:IF R$="" THEN 700
710 IF R$<>"S" AND R$<>"P" AND
    R$<>"U" AND R$<>"R" THEN 70
    0
720 PRINTP$;PP$;CHR$(17);SP$
730 IF R$="S" THEN C=1
740 IF R$="P" THEN C=2
750 IF R$="U" THEN C=3
760 IF R$="R" THEN C=4
770 TY=128+C:GOTO 940
780 :
790 IF A$<>"2" THEN 830
800 PRINTP$;PP$;SP$;P$;PP$;"CAN
    CELLAZIONE FILE"
810 TY=0:GOTO 940
820 :
830 IF A$<>"3" THEN 880
840 IF TY AND 64 THEN RETURN
850 PRINTP$;PP$;SP$;P$;PP$;"PRO
    TEZIONE FILE"
860 TY=TY OR 64:GOTO 940
870 :
880 IF A$<>"4" THEN RETURN
890 IF (TY AND 64)=0 THEN RETUR
    N
900 TY=(TY AND 255)-64
910 PRINTP$;PP$;SP$;P$;PP$;"SPR
    OTEZIONE FILE"
920 :
930 REM  ROUTINE DI RISCRITTURA
     SU DISCO
940 PRINT#15,"B-P 2 ";2+32*I
950 PRINT#2,CHR$(TY);
960 PRINT#15,"B-P 2 0"
970 PRINT#15,"U2 2 0 18 ";S:PRI
    NTP$;PP$;SP$;:RETURN
```

distributore autorizzato COMMODORE

Iscriviti subito all'AMIGA NIWA Club

A tutti gli acquirenti di un P.C. AMIGA (2.200.000 + IVA) in regalo 2 pacchetti software originali e la tessera AMIGA NIWA CLUB.

Vasta biblioteca software già disponibile.

Inoltre la NIWA vi propone biblioteca software per Atari 520/1040-ST e per il vostro C/64-C128:

Dischi 3 1/3 - 1/2	a partire da	**L. 3.800**
SPEEDDOS C64/C128:	il migliore e più collaudato velocizzatore, copia del disco, anche protetto, in 21 secondi, legge i 202 blocchi in 10 secondi, tasti funzione, hardcopy, comandi al D.O.S. diretti	**L. 65.000**
Fast Load Cartridge C64/C128:	il più venduto in Italia, semplicissimo da usare, velocizza di 5 volte il tuo drive, utilities varie con reset	**L. 35.000**
	senza reset	**L. 30.000**
Cartridge ISEPIC C64 E SOFTWARE DED:	trasferisce su disco il 90% del tuo software protetto	**L. 50.000**
HACKER Cartridge:	trasferisce il 99% del tuo software protetto da nastro a disco e da disco a disco in soli 4 minuti senza bisogno di conoscenza Linguaggio Macchina e di compattamento; lavora inoltre senza disco di lavoro e può essere attivata a programma già caricato. 128 compatibile (in modo 64)	**L. 80.000**
HACKER-TAPE:	permette di ricassettare qualsiasi tipo di programma precedentemente trattato con HACKER, senza nessun problema di blocchi, leggendo in turbo da disco e scrivendo in turbo su nastro	**L. 45.000**
OFFERTA:	HACKER + HACKER TAPE	**L. 100.000**
FLOPPY DISK:	di tutte le marche a partire da	**L. 1.700 d.s.d.d.**

Da fine ottobre la NIWA si trasferisce nel
Nuovo punto di vendita al dettaglio in V. Buozzi 94 a Sesto S.G. MM Marelli
Abbonamenti Software.
Spedizioni in tutta Italia.
Sconti ai grossisti, club, negozi.
I prezzi si intendono mIVA compresa e spese di spedizione escluse.
Per oridini superiori a L. 200.000 spese postali gratuite.

Via Valdimagna 54
P.O. BOX n. 83
20099 Sesto
San Giovanni (MI)
Tel. 02/2440776

QUALSIASI COMPUTER

Una cassaforte su disco

Come evitare che un vostro programma memorizzato su dischetto sia sottoposto a copie non autorizzate

di Giancarlo Mariani

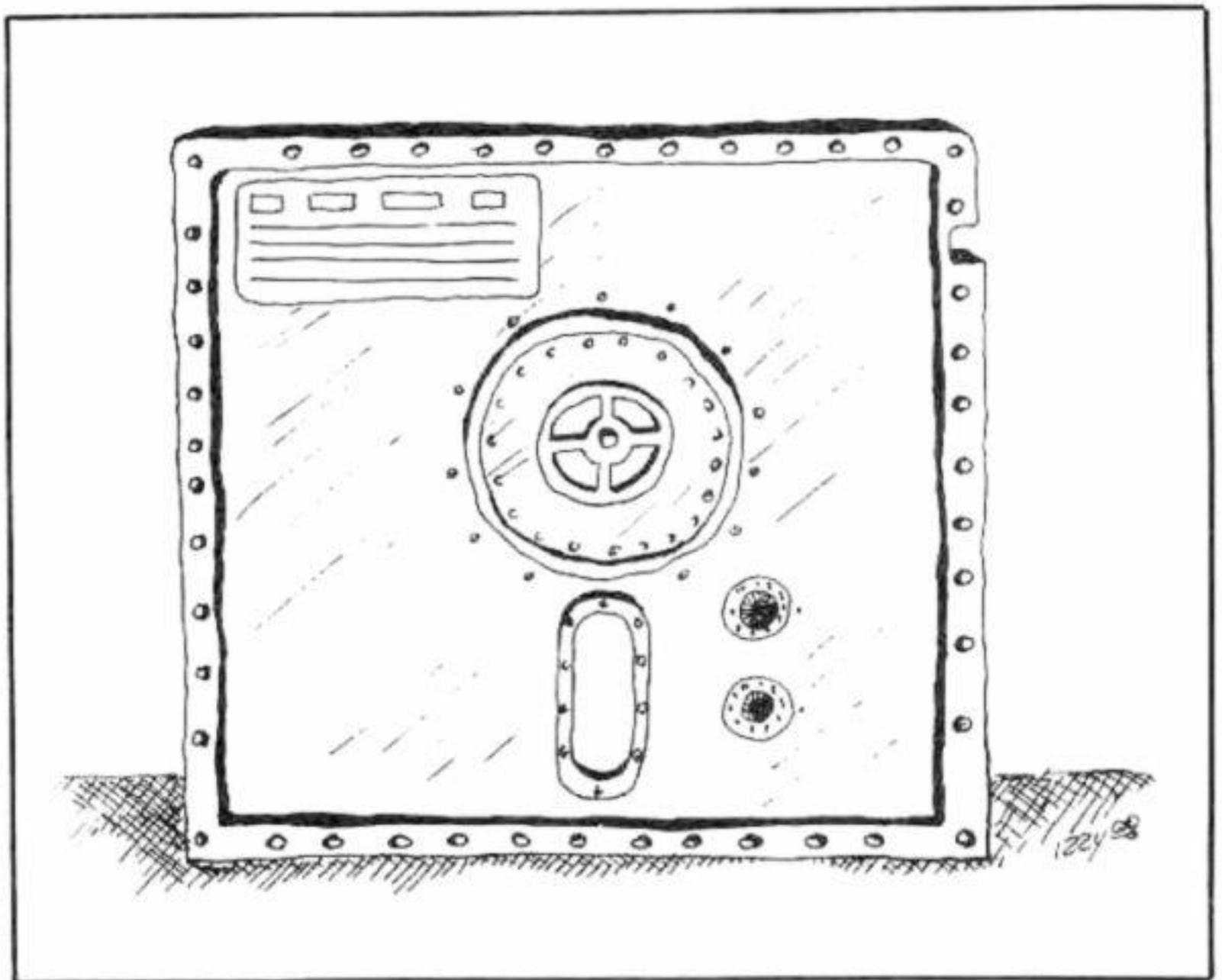
Questo articolo è dedicato ai possessori di un qualsiasi computer Commodore, purchè provvisto di unità a disco.

Spiegare le ragioni che rendono necessaria la protezione di un programma è ormai superfluo; contrariamente a quanto si potrebbe pensare, non solo le Software House possono trovarsi nella necessità di proteggere i loro programmi dalle copie abusive, ma anche voi potreste desiderare che un vostro programma, frutto di sudate ore alla tastiera, non si diffonda... eccessivamente.

Il tipo di protezione che descriveremo nel presente articolo è un po' complicato da effettuare ma ha il notevole pregio di rendere induplicabile un dischetto contenente uno o più programmi.

Esaminando alcuni tipi di protezione presenti in programmi commerciali, studiandone il comportamento ed ascoltando gli strani "rumori" provenienti dal 1541, si arriva ne del disco tale che il drive non riesce a leggerlo correttamente. Questo succede, di solito, con dischetti rovinati fisicamente, oppure non formattati o formattati per macchine diverse da quella per cui andrebbero usati.

Un errore su disco può capitare anche accidentalmente, toccando con le mani la sua superficie, riscaldandolo troppo nel dimenticarlo alla luce del sole o magnetizzandolo me- alla conclusione che il programma "deve", ad un certo punto, controllare qualcosa sul disco, magari uno speciale dato, e se questo non è presente (come nel caso delle copie pira-

ta) il programma si rifiuta di funzionare, resettando la macchina.

In effetti non è un dato ciò che viene controllato sul dischetto, ma un errore! Tutti i possessori di un'unità a dischi dovrebbero sapere che cosa è un "errore".

In pratica si tratta di una alterazio- diante calamite; oppure può essere messo di proposito, proprio per evitare la duplicazione.

La presenza di un errore, infatti, riesce ad arrestare i normali programmi di copia, rendendo impossibile (o quasi) la corretta duplicazione del floppy.

Esistono vari tipi di errore, tra cui quello di scrittura, di lettura, oppure di identificatore.

L'identificatore (ID) di un dischetto è quella stringa di due caratteri da fornire all'atto della formattazione o riformattazione del dischetto (comando OPEN1,8,15,"N0:diskname,ID") e che serve esclusivamente all'unità floppy per distinguere un disco da un altro. Ad ogni disco formattato dovrebbe essere assegnata una ID diversa, per evitare che l'unità vada in "confusione", danneggiando i contenuti del dischetto stesso.

L'identificatore, durante la formattazione, viene trascritto su ciascun settore di ogni traccia del disco. Se ne deduce che se riuscissimo a modifi-

carne qualcuno, il drive, durante la lettura, cadrà in errore bloccando qualsiasi operazione (tra cui la copia) ottenendo lo scopo prefissato.

Ovviamente non basta evitare la duplicazione, ma si dovrà anche far controllare, allo stesso programma, la presenza di tale errore, resettando la macchina in ,aso di riscontro negativo.

Vediamo i passi da seguire per creare un errore di identificatore (errore 29):

1: Formattare un disco nuovo (OPEN15,8,15,"N0:nome,AA": CLOSE15)

2: Impartire nuovamente il comando di formattazione sullo stesso disco, ma con identificatore DIVERSO da quello dato in precedenza (esempio: OPEN2,8,15,"N0:nome,ZZ"). Prima di premere il tasto Return, appoggiate l'orecchio sul drive.

3: Questa è la fase più delicata: ascoltando con grande attenzione i rumori del drive appena inizia la formattazione, dovreste avvertire il rumore di scatti intervallati fra loro con regolarità; ESTRARRE il dischetto subito dopo il secondo scatto, anche se il drive continua a girare.

4: Resettare (o spegnere e riaccendere) il drive.

5: Registrare normalmente, sul disco "trattato", il programma da proteggere.

Come funziona la protezione

Il "trucco" funziona così: formattando la prima volta il disco abbiamo costretto il suo S.O. a scrivere in ogni traccia, e in ogni settore, l'identificatore (nell'esempio di prima: AA).

Con il secondo comando di formattazione (identificatore diverso: ZZ) il drive tenta di ripetere l'operazione con il nuovo ID ma, estraendo il dischetto subito dopo il secondo scatto, fermeremo la formattazione alla seconda traccia, per cui il nuovo ID risulterà trascritto solo sulla prima.

Un secondo accorgimento

Ottenuta la protezione contro la copia integrale del dischetto, faremo ora controllare al programma da proteggere la presenza di tale anomalia in modo da rendere inutile anche la copia "parziale".

La routine per controllare l'errore è proposta, in Basic, in queste pagine. Va inserita all'interno del programma da proteggere, e non fa altro che leggere il primo settore della prima traccia del dischetto, controllando lo stato del disco. Se non riscontra un errore di tipo 29 (Disk ID Mismatch), la routine stampa un messaggio di copia illegale e cancella la memoria del computer.

Per verificare quanto asserito in queste pagine, provate a registrare la routine su di un disco normale, non "trattato", constaterete che essa non funzionerà più a dovere. Non vi saranno inconvenienti se, al contrario, è registrato su disco formattato "a metà".

Chi è intenzionato, però, a copiare illegalmente il programma, potrebbe listarlo, scoprendo (e quindi rimuovendo) le linee che controllano l'errore. Bisognerà quindi trovare il modo di nascondere queste linee, magari compilando il programma, oppure scrivendo una apposita routine in LM, più difficile da identificare, oppure... pensateci voi.

E' da notare che sul dischetto protetto si può registrare anche più di un programma, facendo controllare ad ognuno di essi la presenza della protezione. Si deve avere però l'accortezza di lasciare almeno una cinquantina di blocchi liberi, per evitare che il programma registrato vada a finire proprio sulla traccia protetta.

Le applicazioni intorno a questo tipo di protezione sono infinite e dipendono solo dalla fantasia di chi le utilizza. Si potrebbe, ad esempio, produrre l'errore in più di una traccia, facendole poi controllare tutte; oppure si potrebbe controllare se esistono linee che controllano la presenza di altre linee, che a loro volta... ALT! Fermiamoci qui, altrimenti va a finire che neanche noi saremo più in grado di utilizzare il programma che abbiamo protetto.

```
 10 REM ESEMPIO DI UTILIZZO PRO
    TEZIONE ERRORE 29
 20 REM ROUTINE DA INSERIRE NEL
     PROGRAMMA DA
 30 REM PROTEGGERE CHE VA REGIS
    TRATO POI
 32 REM SUL DISCO PROTETTO
 40 PRINTCHR$(147);"ESEMPIO DI
    PROGRAMMA PROTETTO"
 50 GOSUB  100
 60 IF ERR<>29 THEN PRINT"QUEST
    A E' UNA COPIA PIRATA!":REM
     NEW
 70 PRINT"TUTTO OK":END
100 REM ROUTINE CHE CONTROLLA L
    'ERRORE
110 REM QUESTA, INSIEME ALLA RI
    GA 60
120 REM VA INSERITA NEL PROGRAM
    MA DA PROTEGGERE
130 CLOSE 5:CLOSE 15
140 OPEN 15,8,15:REM APRE IL CA
    NALE DI COMANDO COL DISCO
150 OPEN 5,8,5,"#":REM APRE IL
    CANALE DI COMUNICAZIONE
155 PRINT#15,"B-R:"5;0;1;1:REM
    LEGGE IL BLOCCO 1 DELLA TRA
    CCIA 1
160 INPUT#15,ERR,XZ$,XZ$,XZ$:RE
    M LEGGE L'ERRORE
170 CLOSE 5:CLOSE 15:RETURN
```

QUALSIASI COMPUTER

Su disco tutti i programmi di CCC

Sul N.2 di "Directory" le consuete decine di programmi e, soprattutto, succose novità

di Alessandro de Simone

Inutile dire che l'iniziativa intrapresa sul numero scorso ha avuto un enorme successo.

La novità, stavolta, consiste nel fatto che, su sollecitazione dei lettori, a partire da questo mese inseriremo sul dischetto tutti i listati Basic o L.M. che compaiono sul fascicolo. Su Directory N.2, pertanto, troverete TUTTI i listati di QUESTO numero oltre a TUTTI i listati pubblicati sul numero scorso (N.35).

Allo scopo di risparmiare spazio su disco, precisiamo che le istruzioni dei programmi, come pure le indicazioni sui computer su cui girano, devono esser rintracciate sul fascicolo da cui sono tratti i listati stessi.

Directory N.2

Sicuramente apprezzati saranno i brani musicali presenti, che riproducono musiche classiche (solo per C/64). Ricordiamo che non ci è possibile inserire musiche moderne dal momento che, per la legislazione vigente, bisognerebbe pagare i diritti d'autore alla SIAE (con conseguente incremento del prezzo del disco...).

Il gioco "Mezzogiorno di fuoco" (per C/64) è il videogame italiano della Systems Editoriale che più di ogni altro è stato copiato dai pirati nostrani.

Sono ben quattro le case editrici ladre che hanno scippato il nostro gioco per riproporlo sulle loro cassette ad un prezzo, tra l'altro, superiore al nostro...

Anche "Dynastopoli" è un gioco interessante che va considerato una specie di adventure: siete un giovane juppy che inizia la scalata al successo; riuscirete a far carriera?

Utilissimi, infine, i file che riproducono, con eccezionale fedeltà, voci umane sintetizzate. Si tratta di un micro programma che gestisce il SID del C/64 in modo "totale". Con questi file, e utilizzando la mini routine nei vostri listati personali, potrete far parlare il computer in punti particolari dell'elaborazione. Potrete, tra l'altro, far gridare "Aiuto!" nel caso in cui il vostro omino-sprite si trovi nei guai, oppure fare il conto alla rovescia prima del lancio di un missile o altre cose ancora.

A dispetto della spettacolarità del risultato, ricordiamo che sono in giro da moltissimo tempo opportuni digitalizzatori vocali (hardware). Se qualche lettore ne possiede uno potrà, a sua volta, proporre altre frasi sintetizzate da presentare agli utenti di "Directory".

Partecipazione dei lettori

Centinaia sono le occasioni per partecipare a "Directory".

Prima di inviare il risultato del vostro lavoro vi consigliamo, però, di telefonarci per stabilire se risponde ai requisiti per l'eventuale pubblicazione (tel. 02/84.67.34.8)

Come procurarsi "Directory"

Avvertiamo i lettori che NON è assolutamente possibile inviare i programmi su nastro, per intuibili motivi di economia ed affidabilità del nastro cassetta.

Ogni numero di "Directory" può quindi esser richiesto SOLO su disco inviando L.12000 per ciascun disco oltre a L.3000 (fisse) per le spese di spedizione (indipendenti dal numero di dischi richiesti).

Non ci è possibile inviare materiale contrassegno.

Compilate un normale modulo di C/C postale indirizzando a:

C/C postale N. 37952207
Systems Editoriale
Viale Famagosta, 75
20142 Milano

Non dimenticate di indicare chiaramente, sul retro del modulo (nello spazio indicato con "Causale del versamento") non solo il vostro nominativo completo, ma anche il nome del disco desiderato:

"Directory N.2"

N.B. Per ottenere il materiale ordinato in tempi più ristretti, inviate l'importo a mezzo assegno bancario non trasferibile: le poste italiane non brillano per velocità! (due mesi circa per il recapito di un C/C postale).

DIRECTORY

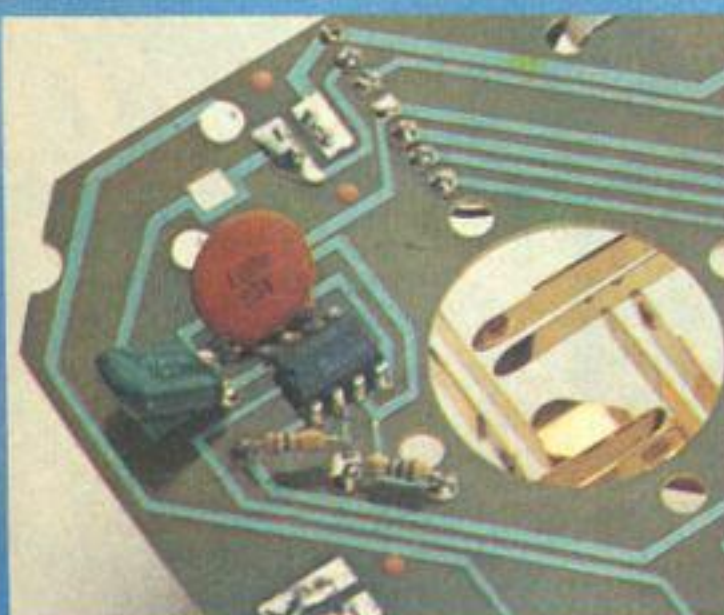
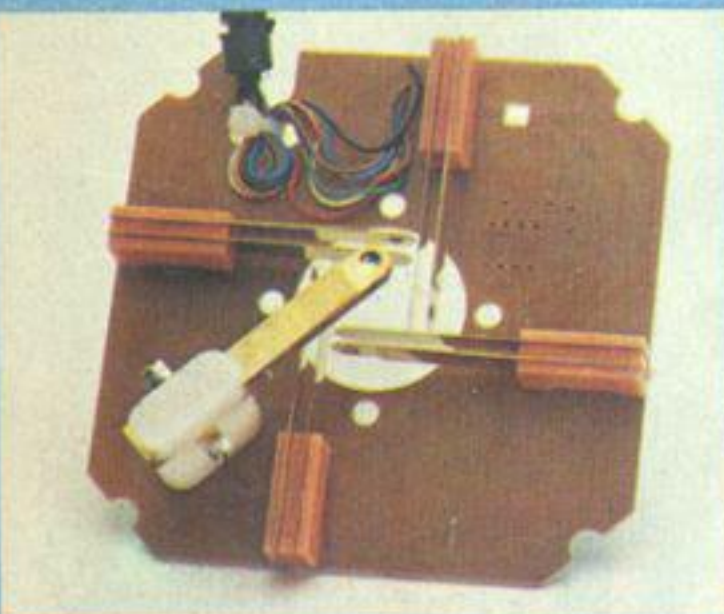
```
0  "DIRECTORY          " D2 2A
2    "REM KILLER PROGR"  PRG
2    "GIOC/AZZ QUATTR"   PRG
5    "GIOC/AZZ TRE"      PRG
2    "GIOC/AZZ DUE"      PRG
1    "GIOCO AZZ.UNO"     PRG
4    "TOT/13 DISPLAY"    PRG
16   "TOT/13 PROGRAMMA"  PRG
6    "PRG MACCHIN/TEMP"  PRG
4    "PRG/SUPERTASTIER"  PRG
3    "PRG SCRAMBLER"     PRG
2    "PRIMO ESEMPIO"     PRG
3    "SECONDO ESEMPIO"   PRG
5    "TERZO ESEMPIO"     PRG
8    "QUARTO ESEMPIO"    PRG
8    "DEMO SPRITE"       PRG
30   "CHAR/EDIT/128"     PRG
9    "SIMUL FROGGER/64"  PRG
4    "DATA DEC-ESA"      PRG
1    "DIMOST.DEC/ESA"    PRG
2    "DATA LOC/CURSOR"   PRG
1    "DEMO LOC/CURS"     PRG
3    "DATA BEEP"         PRG
1    "DEMO BEEP"         PRG
3    "50100 DIRECTORY"   PRG
3    "15100 LAMPEGGIO"   PRG
4    "14900 DEL/WINDW"   PRG
4    "15000 FRAMMENTA"   PRG
3    "PRG AUTOBOOT 128"  PRG
0    "---C.C.C. 36----"  USR
9    "PRG CERCAPAROLA"   PRG
2    "LOTTO 1"           PRG
2    "LOTTO 2"           PRG
7    "LOTTO 3"           PRG
9    "PRG DEDUZIONE"     PRG
2    "CALCIO EFFICIE.1"  PRG
6    "CALCIO EFFICIE.2"  PRG
4    "CALCIO EFFICIE.3"  PRG
17   "PRG ARCHIVIO TOT"  PRG
14   "CORSA CAVALLI"     PRG
4    "GIOCO INNOCENTE."  PRG
1    "DEMO PAUSE 1920"   PRG
4    "PAUSE 1920/962"    PRG
3    "DEMO COLLISIONE"   PRG
5    "COLL/21840/919"    PRG
10   "SP/TOOL/21474/83"  PRG
2    "DEMO SPR/TOOL"     PRG
7    "SPR/GEN/21261"     PRG
4    "DEMO SPR/DEFINIT"  PRG
5    "1.DISKMON/OGGETT"  PRG
24   "1.DISKMON/SOURCE"  PRG
9    "DISK MANAGER"      PRG
4    "6 PR/DIRECTORY"    PRG
3    "5 ID 5 CARATTERI"  PRG
3    "4 NOME DISCO"      PRG
2    "3 PROTEZ DISK"     PRG
2    "2 DISPLAY DIRECT"  PRG
4    "1 MEMORIZZA DIRE"  PRG
3    "ERROR DISK N.29"   PRG
3    "156 CARTES/POLAR"  PRG
2    "155 CON.LOGARIT."  PRG
3    "154 CONVERSIONI"   PRG
3    "153 SEMPL/FRAZ."   PRG
3    "152 DIV/N/DECIM."  PRG
0    "--ALTRI PROGR.--"  USR
1    "ISTRUZIONI MUSIC"  PRG
9    "KOMMUNIST MUSIK"   PRG
9    "ZARATHUSTRA"       PRG
9    "PRIMAVERA"         PRG
9    "40 MOZART"         PRG
9    "TOCCATA E FUGA"    PRG
1    "ISTRUZIONI VOCE"   PRG
9    "VOCE 1"            PRG
9    "VOCE 2"            PRG
9    "VOCE 3"            PRG
9    "VOCE 4"            PRG
9    "VOCE 5"            PRG
9    "VOCE 6"            PRG
9    "VOCE 7"            PRG
103  "DINASTOPOLI"       PRG
100  "MEZZOG.FUOC"       PRG
9    "XYZ"               PRG
0  BLOCKS FREE.
```

A COLPO SICURO

SURESHOT

A colpo sicuro per la forma ergonomica dell'impugnatura, per la struttura ANTIURTO, per il suo snodo in gomma con scorrimenti in teflon che rendono dolce il comando e immediato il ritorno a zero.

GoldSHOT

Aggiungi un gioiello al tuo computer con un joystick a contatti lamellari ORO SU ORO 24 carati. Una resistenza nell'ordine dei milliohm garantisce prontezza assoluta nei comandi direzionali e di sparo.

FIRESHOT

Per una sempre più eccitante azione di gioco, un dispositivo elettronico di FUOCO CONTINUO, è attivabile in tutti i giochi a software compatibile. Anche sul fireshot trovi i contatti oro su oro o se vuoi a MICROSWITCH.

THE PROFESSIONAL GAME

Prodotto in Italia da BREMBOTECNICA - Via M. Liberta 20 - 20090 SEGRATE, tel. 2131941 - Tx. 313576 RACOMI

Il vertice di Mosca e altro ancora...

Anteprima sui programmi di Software Club

di Michele Maggi

E' in edicola la nuova cassetta della Systems Editoriale che, come sempre, contiene numerosi programmi dedicati ai computer più diffusi: C64/128, C16-Plus/4, Vic 20, Spectrum e MSX.

Ecco, in breve, la trama dei videogame e l'argomento delle utility per i computer Commodore.

C64/C128

Moscow Summit

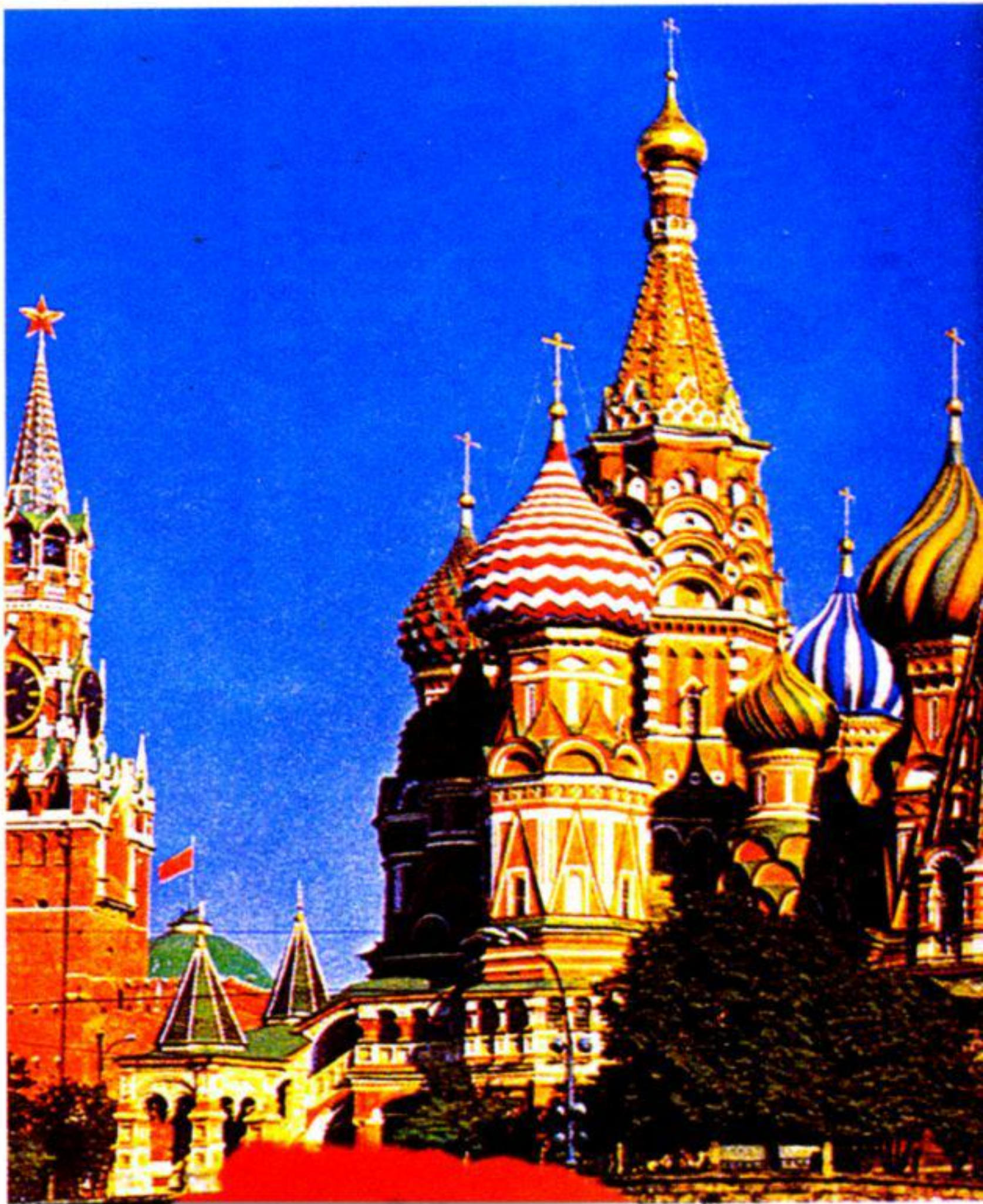
Per i patiti del genere "adventure" una avventura ispirata alle più famose spy story ambientata a Mosca, che vedrà il giocatore calato nella parte di un agente segreto alle prese con i servizi segreti di tutto il mondo.

Si tratta di una avventura interamente descrittiva anzichè semigrafica in quanto le immagini si devono formare nella mente del giocatore invece che sul video, in modo da poter essere maggiormente conivolti nei vari eventi.

La routine di input è stata particolarmente curata in modo da accettare frasi molto complesse e in prima persona, ad esempio:

Scendo dalla macchina e seguo l'uomo

Siamo sicuri che Moscow Summit, proprio come un buon libro, vi farà compagnia a lungo, anche perchè all'interno del programma è prevista l'opzione SAVE che permette di sal-

vare su supporto magnetico la posizione, in modo da poterla riprendere in qualsiasi momento senza dover cominciare tutto da capo.

Poing

Dalle avventure ad un gioco d'azione che ha per protagonista una pallina che, balzando da una piattaforma aerea all'altra, deve distruggere le altre palline presenti saltandovi sopra facendo però molta attenzione a calcolare bene le distanze per evitare di cadere del vuoto con conseguente ingloriosa fine...

Attenzione anche a quegli strani esseri, simili ad amebe... volanti, il cui contatto si rivela letale.

Derby

Un classico gioco ispirato alle corse di cavalli in cui fino a 10 giocatori possono scommettere sul loro beniamino (sia vincente che piazzato).

Dopo le scommesse inizierà la corsa vera e propria che gli scommettitori seguiranno certamente con il fiato sospeso.

Al termine della corsa verranno elencate le somme (vinte o perse) per ogni scommettitore, quindi si passerà ad un'altra corsa.

Around the world

Con il Joy in porta 2 eccoti alla guida di una mongolfiera che, andando in giro per il mondo, deve cercare di rifornirsi di energia sparando sui "distributori" riconoscibili dalla scritta "take me" onde poter percorrere più strada possibile.

Purtroppo andare in giro per il mondo comporta anche vari inconvenienti per cui la nostra mongolfiera dovrà evitare tutti quegli oggetti che cercheranno di colpirla per farle terminare prematuramente il suo viaggio.

Leasing

Un programma di semplice utilizzo in grado di calcolare il tasso percentuale dell'interesse realmente pagato a seguito di un'operazione di leasing.

E' un programma che si rivelerà certamente utile a chi ha intenzione di effettuare investimenti con pagamento rateizzato.

C16/PLUS 4

The wall

Il classico gioco da bar, Breakout, in versione domestica per avvincenti sfide fra amici.

Scopo del gioco è di colpire la pallina mandandola contro il muro per distruggere il maggior numero di mattoni.

Per il controllo della racchetta, usare i tasti ⟨ e ⟩.

Derby

Come per la versione per C64, anche in questo caso abbiamo avvincenti corse di cavalli con annesse scommesse.

Space 1999

Un altro gioco della serie "Space invaders" che vi vede protagonisti nella strenua difesa del vostro pianeta.

Vic 20

Crash

Siete alla guida di una automobile che, all'interno di un labirinto, deve raccogliere il maggior numero di diamanti facendo attenzione a non entrare in collisione con le altre automobili.

Protector

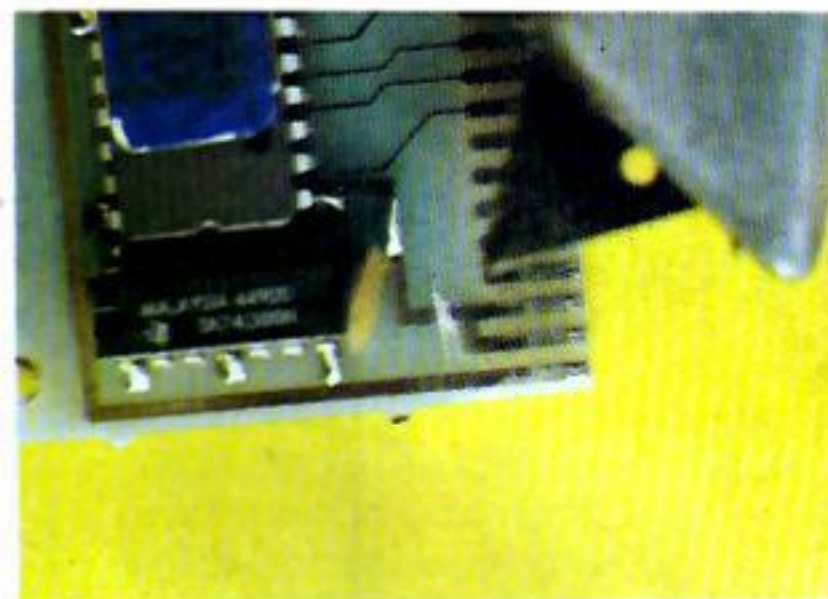
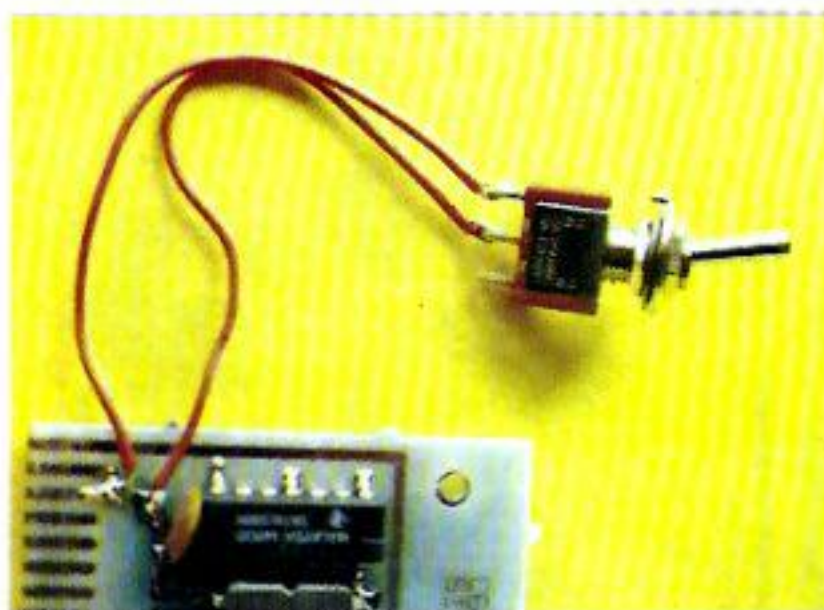
Interamente realizzato in Linguaggio Macchina, si propone di offrire la possibilità a tutti di proteggere, in modo sicuro ed affidabile, qualsiasi tipo di programma, sia esso in Basic o in Linguaggio Macchina.

Protector necessita per poter essere utilizzato dell'aggiunta di una espansione di memoria da 8 o più K Ram.

Il programma implementa, in formato solo diretto, due nuovi comandi dedicati ad una simile operazione di SAVE esclusivamente, si badi bene, su nastro, e nel formato comunemente denominato "turbo".

Un programma che darà filo da torcere anche ai "pirati" più agguerriti...

Come svegliare (o addormentare) una cartuccia

Un interruttore, un paio di saldature ed una mini-operazione chirurgica per abilitare o disabilitare una cartuccia senza rimuoverla dal suo alloggiamento

di Alessandro de Simone

Sul N.31 di Commodore Computer Club, ed esattamente nell'articolo "Come ti sproteggo il cartridge", veniva presentato un progetto in cui si consigliava di manomettere il calcolatore allo scopo di aggiungere un pulsante di Reset ed un interruttore che fornisse tensione, o meno, alla cartuccia.

Alcuni lettori avrebbero voluto realizzare le modifiche descritte ma, preoccupati di "deprezzare" il computer, si sono astenuti dal farlo, rinunciando a malincuore alla comoda possibilità.

Nelle foto di queste pagine suggeriamo, come è possibile vedere, di posizionare interruttore e pulsante di Reset non sul computer ma sulla cartuccia stessa che, di norma, è sufficientemente vuota per ospitare sia l'uno che l'altro.

E' ovvio che la pista di rame collegata ai +5 Volt (pin 2 e 3) deve esssere interrotta con un taglierino e collegata ai due "poli" dell'interruttore. Il pulsante di Reset (del tipo normalmente aperto) va collegato tra il pin "C" (lato inferiore del circuito stampato) e la massa.

Il caso del C/128

Il semplice accorgimento descritto consente di tenere in modo permanente una cartuccia sul retro di un C/64 e di attivarla solo se l'interruttore fornisce corrente al cartridge.

Da quanto detto sembrerebbe che inserendo una cartuccia per C/64 sul retro di un C/128, e tenendo l'interruttore aperto (posizione di "spento") al momento dell'accensione del computer, questo si debba configurare in modo C/128.

Purtroppo ciò non accade perchè le inizializzazioni presenti nel calcolatore non si limitano a verificare la presenza dei caratteri codice nelle locazioni da 32768 in poi (vedi articolo "Come disabilitare il tasto di Reset" CCC N.30).

Nel C/128, infatti, la configurazione iniziale si stabilisce anche in base allo "stato" del pin N.9 (Exrom)).

Se, dunque, volete utilizzare una cartuccia per C/64 senza toglierla dal retro del C/128, dovete inserire l'interruttore tra le estremità dell'interruzione da praticare sulla pista N.9 anzichè sulle N.2-3.

La tensione di 5 Volt, quindi, alimenta permanentemente il cartridge, anche se questo non è utilizzato.

Alcune cartucce, addirittura, utilizzano la pista N.8 (Game) per comunicare al calcolatore la propria identità.

Chi possiede un C/128, in definitiva, e vuole apportare la semplice modifica H/W ad una cartuccia per C/64, deve interrompere la pista 9 (oppure la 8, se il procedimento descritto non porta ai risultati desiderati).

22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1

Z Y X W V U T S R P N M L K J H F E D C B A

PORTA ESPANSIONE

Lasciando la cartuccia sul retro si otterranno le tre configurazioni possibili come segue:

• **Modo C/128**
Escludere l'interruttore e accendere il computer.

• **Modo 64 "normale"**
Escludere l'interruttore e accendere il computer tenendo premuto il tasto Commodore.

• **Modo 64 Cartuccia**
Inserire l'interruttore e accendere il computer.

Se l'interruttore è collegato con la pista +5 Volt, il C/128 si sveglierà sempre in modo 64 e per utilizzarlo in modo 128 (oppure CP/M) sarà necessario rimuovere la cartuccia prima dell'accensione.

Archivio Totocalcio

Come realizzare un archivio per effettuare ricerche o statistiche alla velocità del linguaggio macchina

di Antonio Pastorelli

Vi è mai capitato, nel compilare una schedina, di domandarvi se e quante volte le colonne pazientemente preperate sono già risultate vincenti, oppure quante volte una determinata sequenza di segni è comparsa negli ultimi anni?

Il programma che proponiamo ha lo scopo di aiutarvi nella compilazione delle schedine esaminando, appunto, se (e quante volte) una certa sequenza si è già verificata in passato e se, di conseguenza, ha una certa probabilità di ripresentarsi, o meno.

Basterà inserire pazientemente, tramite tastiera, tutte le colonne risultate vincenti a partire... dall'anno che volete: il numero di "giornate" del campionato, valide ai fini del Totocalcio, sono infatti poche decine per ogni anno.

Il problema principale del reperimento delle informazioni si risolve facilmente recandosi alla biblioteca della vostra Città che, certamente, avrà la raccolta delle annate passate di un qualsiasi quotidiano; sfogliando la cronaca sportiva dei lunedì, e trascrivendo le colonne voncenti, tornati a casa le riverserete nella memoria del computer in modo da realizzare un archivio personale da consultare al momento opportuno.

Come utilizzare il programma

Trascritto, registrato e verificato il programma, non appena impartite il Run compare la domanda "Ti sei ricordato di cambiare nome all'archivio?" di vitale importanza per chi utilizza il drive.

Rispondendo "N", infatti, verrà visualizzata la linea 650 all'interno della quale compare, tra virgolette, il nome "Arch". La prima volta che usate

il programma potete lasciare il nome inalterato; se, al contrario, avrete già memorizzato un archivio (dal nome, appunto, "Arch") dovete cambiare il suddetto nome dal momento che, su uno stesso disco, non è possibile scrivere due file con lo stesso nome. Chi utilizza il registratore, invece, non avrà di questi problemi e alla domanda di prima potrà sempre rispondere affermativamente (S).

Effettuato, dunque, l'eventuale cambio di nome, impartite nuovamente Run e, comparsa nuovamente la domanda di prima, digitate "S".

Alla nuova domanda sul computer utilizzato rispondete premendo il tasto "1" (C-16) oppure "2" (C/64 o Plus/4). A causa della diversa quantità di memoria l'archivio potrà, infatti, assumere dimensioni differenti ma, in ogni caso, più che sufficienti per i nostri scopi, se considerate che mediamente, ogni anno si gioca, al

> Nel digitare righe di programma basic che contengono istruzioni DATA, è piuttosto facile incorrere in errori di digitazione. Supponiamo che un'ipotetica linea basic numerata con 1200 debba contenere i tre valori: 123, 456, 789. Ecco alcuni esempi di errori più frequentemente commessi:
>
> 1200 DATA,123,456,789
>
> C'è una virgola dopo la parola "DATA". I dati letti dal computer sono, in questo caso, quattro: 0, 123, 456, 789. Se, infatti, non figura alcun carattere dopo l'istruzione DATA, automaticamente viene assunto il valore nullo (0).
>
> 1200 DATA 123,456,789,
>
> In questo caso, dopo il numero 789, il computer, grazie alla presenza della virgola erroneamente inserita, "crede" che ci sia un altro valore e, non trovandolo, lo assume come nullo (0).
>
> 1200 DATA 1234,56,789
>
> La virgola è posizionata male, vale a dire dopo il carattere "4" e non dopo il carattere "3". Il computer non può sapere se il valore esatto è 123 oppure 1234 e individuare un errore, in questo caso, risulta piuttosto laborioso.

massimo, una quarantina di concorsi Totocalcio (52 l'anno per l'Enalotto). Le dimensioni massime, comunque, sono:

* per C/16: 500 colonne
* per C/64 e Plus/4: 800 colonne

Dopo una breve attesa, ecco finalmente il menu:

1/ Aggiornamento
2/ Lettura
3/ Fine lavoro

Supponendo di usare il programma per la prima volta, dovrete premere il tasto "1" che propone, a sua volta, un'altra domanda:
Ex-novo o Continuazione?

Inutile dire che usando il programma per la prima volta dovrete digitare "E" (ex-novo) e, subito dopo, inserire la data relativa alla prima colonna da archiviare facendo in modo che sia di sei caratteri; esempio: 10 febbraio 1950, verrà accettata soltanto come 100250.

Finalmente compare la griglia, sovrastata dal numero d'ordine, in cui digiterete le colonne vincenti, una alla volta.

Nell'introduzione delle colonne sarete facilitati dalla presenza della griglia del tutto simile a quella che abitualmente compare nelle schedine del Totocalcio.

Nel caso commettiate un errore di battitura, poco male: completate i 13 pronostici e, alla domanda di conferma, rispondete negativamente. La colonna appena inserita verrà ignorata e a conferma di ciò noterete che il numero d'ordine (in alto sul video) è rimasto immutato.

Dopo ogni risposta affermativa verrà chiesto se intendete continuare o se volete registrare l'archivio digitato, per riprenderlo, magari, successivamente.

Se ritenete di esservi affaticati troppo, premendo "S" verrà chiesta la data relativa all'ultima colonna inserita, in modo che la volta successiva sarete facilitati a rintracciare la colonna che dovte digitare per incrementare l'archivio.

Non appena il file-archivio è registrato comparirà, probabilmente, un Syntax Error. Niente paura: il fenomeno è dovuto all'alterazione dei puntatori e tale operazione fa "impazzire" il computer.

Vi siete ricordati, prima, di registrare il programma secondo le nostre raccomandazioni? Noi speriamo di sì, perchè, dopo la regitrazione dell'archivio, il listato viene automaticamente cancellato...

A questo punto è bene spegnere il computer e riaccenderlo per evitare continue segnalazioni di "Out of memory Error".

ATTENZIONE: quando si riprende un lavoro già iniziato (e, comunque, per utilizzare l'archivio registrato su supporto magnetico) bisogna attenersi scrupolosamente alle seguenti istruzioni:

1/ digitare (caso del nastro):
LOAD"nome archivio",1,1
oppure (disco):
LOAD"nome archivio",8,1
2/ digitare NEW e premere il tasto Return
3/ caricare il programma e dare il solito Run

Al posto di "nome archivio" digiterete, ovviamente, il nome realmente assegnato alla riga 650 ricordata all'inizio del presente paragrafo.

Per continuare ad archiviare, rileggete la prima parte dell'articolo: non dovreste avere difficoltà nel continuare l'inserimento.

Come consultare l'archivio

Per la consultazione dell'archivio, invece, dovrete premere il tasto "2" (Lettura) al momento della comparsa del menu. Vi sono, selezionando tale opzione, due possibilità:

1/ Ricerca sequenza di segni in una specifica posizione all'interno delle colonne vincenti.

2/ Ricerca sequenza di segni senza tener conto della posizione all'interno delle colonne vincenti

Vediamo in che modo viene attivata la ricerca nei due casi.

Supponiamo che il nostro archivio

sia composto di sole quattro colonne, ed esattamente:

a/ 111,111,111,1X1X
b/ 111,111,XXX,1X11
c/ 111,111,111,1111
d/ 111,2X2,XXX,1111

Supponiamo di voler sapere in quante colonne vincenti compare la sequenza "111" nelle righe 4,5 e 6. Dall'esame del nostro mini archivio notiamo che posseggono questa caratteristica le colonne a, b, c (la d contiene, nella posizione accennata, 2X2).

Premeremo, pertanto, il tasto "0" (zero) per tre volte dal momento che le prime tre righe non interessano. Batteremo, invece, tre volte il tasto "1" in corrispondenza delle righe 4,5,6. Poichè non interessa il contenuto delle rimanenti, premeremo il tasto 0 fino alla 13ma riga e, alla domanda di conferma, il tasto Return. La risposta, ovviamente, è 3. Se compare una risposta diversa vuol dire che avete sbagliato a digitare il programma dalla rivista.

Vediamo, ora, di sfruttare la seconda opzione. Con questa sarà possibile sapere quante colonne presentano una successione di segni indipendentemente dalla loro effettiva posizione all'interno della colonna.

Se, per esempio, vogliamo sapere in quante colonne la successione "X1" si è verificata, dando uno sguardo al mini archivio di prima, noteremo che tale sequenza è presente nella colonna "a" (righe 11-12) nella "b" (righe 9-10 e 11-12) e nella "d" (9-10). La risposta, quindi, sarà: 3. La colonna "b", ovviamente, viene conteggiata una sola volta benchè la presenza sia doppia. Ciò è stato fatto apposta, per evitare statistiche falsate da ripetizioni nella stessa colonna.

Per comunicare al programma l'intenzione di verificare la presenza di successioni, dovremo premere la sequenza desiderata, indipendentemente dalla posizione nella griglia che compare, e poi premere il tasto Return.

Nel caso appena esaminato, in altre parole, premerete direttamente "X", "1" e Return per sapere l'informazione desiderata.

COMPRAMI E TI FARO' UN REGALO!

Acquistando due scatole di Flexible Disk MEMOREX puoi chiedere subito un omaggio simpatico, originale e utile:

l'orologio impermeabile sport-time MEMOREX con il portamonete da polso.

è importante scegli

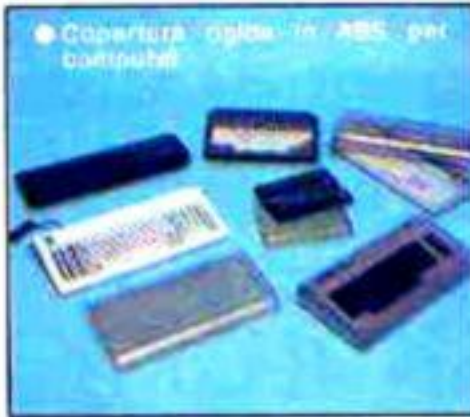
MEMOREX

A Burroughs Company

adverteam

Autorizzazione Ministeriale in corso

```
  1 REM ARCHIVIO TOTOCALCIO
  2 REM PER C/64, C/16, PLUS/4
  3 :
  5 POKE 55,184:POKE 56,34:CLR
 10 PRINT"[CLEAR]TI SEI RICORDA
    TO DI CAMBIARE IL NOME     D
    ELL'ARCHIVIO (S/N)?"
 20 GET A$:IF A$="S" THEN 50
 30 IF A$="N" THEN LIST 650:PRI
    NT TAB(10)"[UP]↑":LIST 660:
    PRINT TAB(10)"[UP]↑":END
 40 GOTO 20
 50 PRINT"[CLEAR]STAI USANDO IL
    ":PRINT"[2 DOWN]1)C-16":PRI
    NT"2)C-64 O PLUS/4"
 60 GET A$:IF A$<"1" OR A$>"2"
    THEN 60
 70 IF A$="2" THEN FI=19517:POK
    E 8926,0
 80 IF A$="1" THEN FI=15617:POK
    E 8926,1
 90 PRINT"[CLEAR]ATTENDI."
100 X1$=CHR$(192):X2$=CHR$(178)
    :X3$=CHR$(177):X4$=CHR$(171
    ):X5$=CHR$(179)
101 X6$=CHR$(219)
110 X7$=CHR$(213):X8$=CHR$(201)
    :X9$=CHR$(202):XA$=CHR$(203
    ):XB$=CHR$(221)
120 FOR J=1 TO 20:B$=B$+CHR$(17
    ):NEXT:DIM B(13):GOSUB 940
130 RT=0:PRINT"[CLEAR][RVS]  AR
    CHIVIO COLONNE VINCENTI TOT
    OCALCIO   "
140 PRINT"[3 DOWN]1)AGGIORNAMEN
    TO":PRINT"2)LETTURA":PRINT"
    3)FINE LAVORO"
150 GET A$:IF A$="1" THEN 190
160 IF A$="2" THEN 670
170 IF A$="3" THEN PRINT"[CLEAR
    ]";:NEW
180 GOTO 150
190 PRINT"[CLEAR][RVS]E[RVOFF]X
    -NOVO O [RVS]C[RVOFF]ONTINU
    AZIONE?"
200 GET A$:IF A$="E" THEN A=914
    4:POKE 8915,184:POKE 8916,3
    5:GOTO 900
210 IF A$="C" THEN 230
220 GOTO 200
230 PRINT"[CLEAR]DATA INIZIO:";
    :FOR J=1 TO 6:PRINTCHR$(PEE
    K(8888+J));:NEXT:PRINT
240 PRINT"DATA ULTIMA COLONNA:"
    ;:FOR J=1 TO 6:PRINTCHR$(PE
    EK(8894+J));:NEXT:PRINT
250 A=PEEK(8915)+256*PEEK(8916)
    :PRINT"[DOWN]COLONNE INSERI
    TE:";(A-9144)/13:B=800
260 IF PEEK(8926)=1 THEN B=500
270 PRINT"[DOWN]PUOI INSERIRNE:
    "B-(A-9144)/13
280 GET A$:IF A$="" THEN 280
290 PRINT"[CLEAR][RVS]          A
    G G I O R N A M E N T O
         [HOME]"(A-9144)/13+1
300 PRINT"[HOME][DOWN]"X7$X1$X1
    $X1$X2$X1$X8$:FOR J=1 TO 3:
    FOR X=1 TO 3:PRINTXB$"     "X
    B$" "XB$
301 NEXT
310 PRINTX4$X1$X1$X1$X6$X1$X5$:
    NEXT
320 FOR J=1 TO 4:PRINTXB$"     "X
    B$" "XB$:NEXT:PRINTX9$X1$X1
    $X1$X3$X1$XA$
330 PRINT"[HOME][DOWN]":X=0:FOR
     J=1 TO 13:X=X+1:IF X>3 AND
     J<12 THEN X=1:PRINT"[DOWN]
    ";
340 PRINT TAB(4-LEN(STR$(J)));J
    :NEXT
350 PRINT"[HOME][DOWN]":X=0:FOR
     J=1 TO 13:X=X+1:IF X>3 AND
     J<12 THEN X=1:PRINT"[DOWN]
    ";
360 GET A$:IF A$=CHR$(13) AND R
    T=2 THEN A$="0":B(J)=0:GOTO
     440
370 IF A$="0" AND RT=1 THEN B(J
    )=0:A$=" ":GOTO 420
380 IF A$="1" THEN B(J)=1:GOTO
    420
390 IF A$="X" THEN B(J)=2:GOTO
    420
400 IF A$="2" THEN B(J)=3:GOTO
    420
410 GOTO 360
420 A=A+1:IF A>19543 THEN PRINT
    "[CLEAR]NON PUOI INSERIRE A
    LTRE COLONNE.":GOTO 860
430 PRINT TAB(5);"[RVS]";A$:NEX
```

```
    T
440 PRINT"[HOME]"B$;"CONFERMI (
    S/N)?"
450 GET A$:IF A$="N" THEN 480
460 IF A$="S" THEN 500
470 GOTO 450
480 IF RT THEN PRINT"[CLEAR]":G
    OTO 300
490 A=PEEK(8915)+256*PEEK(8916)
    :GOTO 290
500 IF RT THEN FOR J=1 TO 13:PO
    KE 8900+J,B(J):NEXT:RETURN
510 FOR J=1 TO 13:POKE A-14+J,B
    (J):NEXT
520 POKE 8916,INT(A/256):POKE 8
    915,A-INT(A/256)*256
530 PRINT"[CLEAR][RVS]C[RVOFF]O
    NTINUA INSERIMENTO[RVS]
                  S[RVOFF]ALVA";
540 FOR J=1 TO 40:PRINT"[RVS] "
    ;:NEXT
550 GET A$:IF A$="C" THEN 290
560 IF A$="S" THEN 920
570 GOTO 550
580 PRINT"[RVOFF][2 DOWN]SU [RV
    S]N[RVOFF]ASTRO O [RVS]D[RV
    OFF]ISCO?"
590 GET A$:IF A$="N" THEN POKE
    8927,1:GOTO 620
600 IF A$="D" THEN POKE 8927,8:
    GOTO 640
610 GOTO 590
620 PRINT"[2 DOWN]RIPOSIZIONA I
    L NASTRO E PREMI RETURN....
    "
630 GET A$:IF A$<>CHR$(13) THEN
     630
640 IF PEEK(8926)=1 THEN 660
650 POKE 43,185:POKE 44,34:POKE
     45,88:POKE 46,76:SAVE "ARC
    H",PEEK(8927),1:NEW
660 POKE 43,185:POKE 44,34:POKE
     46,61:POKE 45,28:SAVE "ARC
    H",PEEK(8927),1:NEW
670 PRINT"[CLEAR][RVS]
         L E T T U R A
```

```
      "
680 PRINT"[2 DOWN]1)RICERCA SEQ
    UENZA DI SEGNI IN UNA SPE-
      CIFICA POSIZIONE"
690 PRINT"[2 DOWN]2)RICERCA SEQ
    UENZA DI SEGNI IN QUALSIASI
      POSIZIONE"
700 GET A$:IF A$="1" THEN RT=1:
    PRINT"[CLEAR]";:GOSUB 300:G
    OTO 730
710 IF A$="2" THEN RT=2:PRINT"[
    CLEAR]";:GOSUB 300:GOTO 760
720 GOTO 700
730 FOR J=1 TO 13:IF B(J)<>0 TH
    EN 750
740 NEXT:GOTO 130
750 GOSUB 820:SYS9063:GOTO 790
760 RT=0:VA=0:FOR J=1 TO 13:IF
    B(J)=0 THEN GOSUB 850:GOTO
    780
770 POKE 8900+J,B(J):VA=VA+1:NE
    XT
780 POKE 8914,VA:POKE 8927,14-V
    A:GOSUB 820:SYS8928
790 PRINT"[2 DOWN]LA SEQUENZA S
    I E' VERIFICATA";PEEK(8921)
    +256*PEEK(8922);"VOLTE."
800 GET A$:IF A$="" THEN 800
810 GOTO 130
820 XX=(PEEK(8915)+256*PEEK(891
    6)-9141)/13
830 POKE 8920,INT(XX/256):POKE
    8919,XX-INT(XX/256)*256
840 POKE 38,184:POKE 39,35:POKE
     8917,0:POKE 8918,0:POKE 89
    21,0:POKE 8922,0:RETURN
850 FOR K=J TO 13:POKE 8900+K,0
    :NEXT:RETURN
860 FOR J=1 TO 1500:NEXT:PRINT"
    [CLEAR]REGISTRI (S/N)?"
870 GET A$:IF A$="N" THEN 130
880 IF A$="S" THEN 580
890 GOTO 870
900 INPUT "DATA INIZIO (6 CAR)"
    ;D$:IF LEN(D$)<>6 THEN PRIN
    T"[UP]";:GOTO 900
910 FOR J=1 TO 6:POKE 8888+J,AS
    C(MID$(D$,J,1)):NEXT:GOTO 2
    90
920 INPUT "[RVOFF]DATA ULTIMA C
    OLONNA (6 CAR)";D$:IF LEN(D
    $)<>6 THEN PRINT"[UP]";:GOT
    O 920
930 FOR J=1 TO 6:POKE 8894+J,AS
    C(MID$(D$,J,1)):NEXT:GOTO 5
    80
940 X=0:FOR J=1 TO 216:READ A:X
    =X+A:POKE 8927+J,A:NEXT
942 IF X<>28177 THEN PRINT"ERRO
    RE NEI DATA...":END
943 RETURN
945 DATA 160,0,162,1,32,48,35,1
    73,222,34,240
950 DATA 46,238,217,34,208,3,23
    8,218,34,238,213,34
955 DATA 208,3,238,214,34,173,2
    13,34,205,215,34,208,9,173
960 DATA 214,34,205,216,34,208,
    1,96
965 DATA 165,38,105,13,144,2,23
    0,39,133,38,76,224,34,232
970 DATA 200,238,223,34,236,223
    ,34,208
975 DATA 6,206,223,34,76,244,34
    ,206,223,34,76,228,34,173
980 DATA 210,34,141,219,34,169,
    0,141
985 DATA 222,34,138,72,138,72,1
    77,38,72,173,210,34,56
990 DATA 237,219,34,170,104,221
    ,197,34,240
1000 DATA 6,104,104,170,168,136,
     96,104,170,232,200,206
1001 DATA 219,34,208,222,169,1,1
     41,222,34
1005 DATA 76,81,35,160,0,162,1,1
     89,196,34,240,7,177,38
1010 DATA 221,196,34,208,44,230,
     38,208,2
1020 DATA 230,39,232,224,14,208,
     233,238,217,34,208,3
1021 DATA 238,218,35,238,213,34,
     208,3,238
1025 DATA 214,34,173,213,34,205,
     215,34,208,207,173,214
1030 DATA 34,205,216,34,208,199,
     96,142
1040 DATA 223,34,169,14,56,237,2
     23,34,24,101,38,144,2,230,3
     9,133,38,76,138,35
```

Intelligenti deduzioni artificiali

Come realizzare, col vostro piccolo home computer, esperimenti nientemeno che di cibernetica!

di Sandro Certi & Franco Toldi

Può una macchina comportarsi in modo intelligente? Attorno a questa domanda si sono sviluppati un grande dibattito e accese dispute, fin dal momento in cui sono stati costruiti i primi calcolatori.

Le risposte, naturalmente, sono state le più varie; più o meno argomentate; più o meno dogmatiche.

Sia pur fortemente ostacolata da cosmici pessimismi, ma anche da infondati superottimismi, l'Intelligenza Artificiale (AI) ha compiuto negli ultimi trenta anni notevoli passi in avanti.

Costituisce oggi, per unanime riconoscimento, il campo più innovativo e promettente della futura ricerca della Computer Science.

L'Intelligenza Artificiale è oggi un terreno di forte investimento per le grandi imprese dell'informatica e aspetto privilegiato dei programmi di ricerca sovvenzionati da Stati e da organismi internazionali.

I progressi in questo campo non sono ancora entusiasmanti, nonostan-

te le notevoli risorse impiegate.

Non sbagliamo certamente nell'affermare che è addirittura difficile definire univocamente che cosa sia l'Intelligenza Artificiale.

Si può però affermare che l'AI è un campo di ricerca mirante a ottenere dai calcolatori un comportamento sempre più intelligente operando sia sul Software che sull'Hardware e sulla loro progettazione e realizzazione.

Per lungo tempo si è cercato di realizzare, con il computer, la migliore simulazione possibile dei procedimenti tipici del pensiero umano.

Ci si è accorti presto che si sapeva ancora poco sul funzionamento della mente umana e sulle metodologie che presiedono alle attività del pensiero intelligente.

Tuttavia, in termini poveri, possiamo azzardare che intelligenza è: raccogliere, associare, scegliere secondo determinati criteri; compiere operazioni logiche di induzione e di deduzione; formarsi opinioni e saperle verificare: "apprendere", insomma, in senso lato.

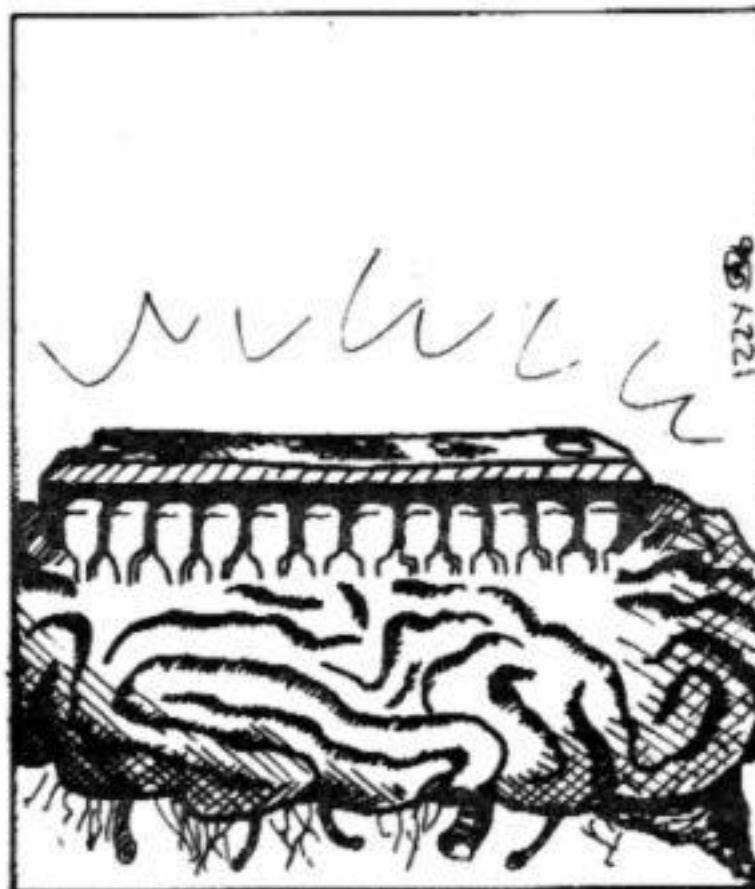
Scoprire e automatizzare meccanismi che presiedono a tali funzioni e applicarli ai programmi dei calcolatori è, appunto, il lavoro dell'AI.

Gli argomenti dell'AI sono vari e molteplici: dalla comprensione automatica del linguaggio naturale scritto e parlato, alla risoluzione automatica di problemi; dalla capacità automatica di apprendimento, al ragionamento logico formale o in base a regole empiriche appositamente determinate; fino ai più complessi "Sistemi Esperti".

Ci proponiamo, in futuro, di presentare ai lettori una serie di programmi, scritti in Basic per i piccoli computer di casa, capaci di attuare in modo semplice le metodologie dei programmi di AI.

Si tratterà di programmi limitati ma funzionanti e, soprattutto, largamente perfezionabili da parte dei lettori.

Un programma che ragiona

Il programma "Operatore Logico" è in grado di compiere automaticamente deduzioni a partire da una serie di premesse, immagazzinate come dati.

Le premesse verranno immesse come proposizioni in linguaggio naturale, pur se in una forma sintattica molto rigida e primitiva.

Il programma provvederà a memorizzare le premesse in forme e modalità adeguate per essere facilmente rielaborate logicamente.

Una volta immagazzinato un certo numero di premesse, si può iniziare a "interrogare" il programma in merito alla verità, o meno, di una qualsiasi proposizione.

A questo punto il programma ha tre diversi possibili comportamenti: se la proposizione di cui si chiede la verifica corrisponde direttamente ad una delle premesse memorizzate, apparirà un messaggio di conferma "Posso confermare".

Se non trova dati corrispondenti alla proposizione in verifica, concluderà "Nessun dato per confermare"; oppure, scorrendo tutte le affermazioni contenute nelle premesse immagazzinate, cerca in premesse diverse un fattore di correlazione che permetta di verificare la proposizione attraverso un ragionamento deduttivo, il cui esito positivo porterà a conferma, altrimenti al messaggio di non conferma.

Come usare il programma

Ricorriamo ora ad un esempio, sia per verificare la corretta trascrizione del programma da parte del lettore, sia per meglio illustrare le modalità di funzionamento.

Digitato il programma, e dato il Run, dopo alcuni secondi compariranno le istruzioni in base alle quali immettere come premesse le seguenti dichiarazioni (non dimenticate "d-"):

d-il falco è un predatore
d-il falco è un uccello
d-l'uccello è un pennuto
d-un pennuto è ricoperto di penne

E' bene ricordare che la sintassi è straordinariamente rigorosa. E' indispensabile, tra l'altro, che i caratteri siano sempre tutti minuscoli oppure tutti maiuscoli; che tra il trattino di "d-" e l'articolo non vi siano spazi; che non vi sia più di uno spazio tra le diverse parole della dichiarazione; che la dichiarazione sia effettuata al singolare e non al plurale ("d-il falco è..." e non: "d-i falchi sono..."). Volendo utilizzare articoli determinativi o indeterminativi, se l'articolo richiede l'apostrofo non vi devono essere spazi tra questo e la parola cui si riferisce.

E' possibile effettuare, comunque, dichiarazioni in vari modi diversi, dal momento che il programma ricorre a particolari "filtri" per esaminare la frase digitata.

E' la stessa cosa, ad esempio, battere:
d-l'uccello è un pennuto
d-l'uccello è pennuto
d-uccello è pennuto

...ma non...

d-uccello pennuto

...pena l'avvertimento "Input non conforme alle regole".

Supponendo, dunque, di aver inserito le quattro dichiarazioni di cui sopra, proviamo a chiedere tutto ciò che è attinente al "Falco", ricorrendo alla sintassi "V-":

v-falco

Il computer risponderà:

Falco è:
Predatore
Uccello

riferendosi, cioè, solo alle due dichiarazioni che direttamente mettono in relazione il falco con alcuni suoi "attributi".

Cerchiamo ora di stabilire un "collegamento" tra le varie dichiarazioni ricorrendo dapprima alla forma interrogativa ("I-") e, subito dopo, ad una nuova richiesta di visualizzazione (V-"). N.B.: [R] significa "Premere il tasto Return":

i-falco è pennuto [R]
Posso confermare
v-falco [R]
falco è:
predatore
uccello
pennuto

Se tentiamo, nuovamente, con:

i-falco ricoperto di penne [R]

il computer dimostrerà di aver acquisito un nuovo dato ad una successiva verifica:

i-falco è ricoperto di penne [R]
posso confermare
v-falco [R]
falco è:
predatore
uccello
pennuto
ricoperto di penne

Le ultime due proposizioni, in definitiva, non erano state direttamente immesse come premesse note, ma la "verità" è stata ricavata collegando tra loro la seconda con la terza premessa esplicita:

il falco è un uccello
l'uccello è un pennuto

dunque si può dedurre che:

il falco è un pennuto

A questo punto la proposizione verificata vera attraverso il processo deduttivo descritto, viene immagazzinata come una nuova premessa esplicita.

Ciò naturalmente permette di proseguire con la verifica di proposizioni avendo a disposizione un numero maggiore di premesse dichiarate o accertate.

Le nostre conoscenze sul meraviglioso mondo degli uccelli non avranno compiuto un grande balzo in avanti, ma resterà di certo la soddisfazione di vedere il computer rispondere con un certo sussiego alle domande, dopo aver compiuto quel procedimento logico detto sillogismo, su cui tanto hanno discusso logici e filosofi.

Attenzione però, il programma è responsabile delle deduzioni che elabora, non della verità assoluta delle premesse che l'utente immette nella sua memoria attraverso frasi dichiarative.

Se immettiamo come premessa la dichiarazione in zoologia palesemente non vera:

d-l'uccello è un mammifero [R]

il programma la assumerà come una premessa logicamente e informaticamente vera, e giudicherà confermata la proposizione:
i-il falco è mammifero [R]
operando una deduzione corretta, ma da una premessa falsa nel mondo della zoologia, anche se considerabile come vera in logica. Di tutto ciò naturalmente non è certo responsabile il computer!

Come funziona il programma

Il programma ha la struttura tipica di tutti i programmi di AI scritti in Basic ed è composto di routine semplici e piuttosto note in letteratura.

La routine "Supervisore" richiama nell'ordine stabilito le routine che svolgono le diverse funzioni e permette l'appropriato scorrimento dell'intero programma.

La prima routine richiamata inizializza l'intero programma dimensionando la matrice quadrata A$ (Y,X) in cui saranno contenuti in forma di stringa i dati ricavati dalle premesse dichiarate dall'utente o ricavate per deduzione dal programma.

La dimensione della matrice è legata al numero di dati che si pensa di immettere contemporaneamente nel programma.

E' bene ricordare che la scansione della matrice parte da A$(0,0), prosegue con A$(0,1), A$(0,2) e così via fino a A$(Y,X).

La matrice viene, all'inizio, interamente riempita assegnando un carattere "spia" (chiocciolina) a tutti i valori, il quale verrà successivamente ogni volta sostituito dai dati che saranno ricavati dalle premesse. La presenza della chiocciolina in una stringa A$(Y,X) indicherà una posizione della matrice non ancora occupata da un dato significativo.

La routine "Messaggio di help" fornisce informazioni sul formato dell'Input.

La routine "Riceve Input" e quella "Smista e elabora input" provvedono ad elaborare la frase digitata dall'utente.

Dapprima esamina i primi due caratteri dell'input ricevuto, ne controlla la coerenza con la sintassi corretta (altrimenti rimanda il controllo al supervisore), li elimina dalla stringa input e attribuisce alla variabile P il valore 1, 2 oppure 3 che indicherà al programma quale funzione l'utente abbia richiesto (immissione di premesse, interrogazione, visualizzazione di dati).

Un secondo compito della routine consiste nell'elaborare la stringa I$ ricevuta in input.

A questo punto il programma prevede una drastica ma necessaria limitazione: le proposizioni dichiarate come premessa o quelle interrogative della loro verificità devono essere espresse in linguaggio italiano naturale, ma in forma rigidamente vincolata.

L'unico formato ammesso è:

(espressione soggetto) è (espressione attributo)

Vediamo ad esempio:

d-il falco è un uccello
i-il falco è un pennuto

Queste sono entrambe formalmente corrette. Nella prima l'espressione soggetto è "Il falco", espressione attributo è "Un uccello". Soggetto e attributo possono essere collegati solamente dal verbo è (nella stringa I$ che riceve l'input i caratteri dell'e-

spressione soggetto devono essere seguiti dai caratteri "spazio è spazio", a loro volta seguiti dai caratteri dell'espressione attributo).

Per la funzione Visualizzazione il formato accettato è:

v-il falco

cioè il programma dopo la V- considera la stringa rimanente come espressione soggetto in base alla routine "Divide Input".

La routine a questo punto elabora la stringa I$ (a cui sono stati già precedentemente tolti i primi due caratteri che indicavano la funzione) e la divide in NO$ che contiene la espressione soggetto che si trova a destra di "spazio" "è" "spazio"; e in AT$ che si trova a sinistra.

Ma il lavoro di "Divide Input" non è ancora terminato. Infatti con le linee di "Toglie articoli" vengono tolti da NO$ e AT$ gli eventuali articoli, allo scopo di memorizzare dati non vincolati dall'articolo che renderebbe più rigida la successiva utilizzazione.

Con un ciclo vengono letti successivamente tutti i possibili articoli immagazzinati nei DATA che, se individuati nelle stringhe NO$ e AT$, vengono eliminati.

Finalmente il controllo ritorna al "Supervisore", il quale in base al valore di P richiama opportunamente la routine richiesta.

La routine "Storage frasi" immagazzina i dati ricavati dalle proposizioni dichiarative immesse dall'utente nella matrice A$(Y,X) secondo un preciso ordine.

Ogni espressione soggetto viene immessa come variabile NO$ nella posizione A$(0,X) non occupata. La corrispondente variabile AT$ viene immessa nella stessa colonna X della matrice al primo valore di Y non occupato (naturalmente a partire da Y=1 perchè in Y=0 sarà collocata sempre la variabile NO$.

Nel caso in cui venga introdotta una dichiarazione attinente ad una espressione soggetto (NO$) già precedentemente immagazzinata con altri attributi (AT$), il nuovo attributo viene immesso nella colonna in cui si trova già la espressione soggetto, nella prima posizione libera (cioè contenente soltanto il carattere "spia").

La Deduzione

La routine "Interrogative" è il vero cuore operativo del programma.

Serve a confermare, o meno, la verità di una proposizione su cui l'utente interroga il computer.

La proposizione arriva alla routine

Una proposta

Il programma può esser la base per sviluppare un gioco basato su deduzioni successive.

Il listato, già nella versione pubblicata, consente di disegnare, tra l'altro, una trama "gialla". Provate a digitare il seguente banalissimo esempio:

d-la moglie è uccisa [R]
d-il marito è geloso [R]
d-il geloso è omicida [R]

In seguito cancellate lo schermo e, chiamato un familiare, dite che, del "caso", è dato di sapere soltanto che una donna è stata assassinata e bisogna risalire al colpevole.

Il familiare, ovviamente, all'inizio è costretto ad andare alla cieca e digiterà, magari:

v-donna [R]
v-morta [R]
v-fidanzata [R]

...ottenendo, come risposta, il laconico messaggio "Nessun dato disponibile". Prima o poi, però, penserà a digitare:

v-moglie

la cui risposta ("uccisa") suggerirà di richiedere notizie sul marito. La presnza di dati (Il marito è geloso) indurrà alla richiesta di frasi del tipo:

i-il marito è assassino [R]

finchè non si perverrà alla necessaria...:

i-il marito è omicida [R]

...oppure...

v-geloso [R]

...seguito, finalmente, da...

i-il marito è omicida [R]

L'esempio riportato, ovviamente, è solo un cenno sulle notevoli possibilità di sviluppare un gioco decisamente più complesso.

Aggiungendo al programma, infatti, una routine per memorizzare su supporto magnetico la matrice-stringa quadrata (ed una per richiamarla in seguito), eliminando la possibilità di fare dichiarazioni, e lasciando disponibili soltanto quelle di interrogazione e visualizzazione, è possibile generare una trama poliziesca in cui l'utente, interrogando il calcolatore, deve individuare l'autore di un delitto.

I lettori più volenterosi sono pertanto invitati a modificare il listato in modo da render possibile la creazione di un gioco, la sua memorizzazione ed il successivo "richiamo" del file che consente di giocare ad un nuovo tipo di "Adventure".

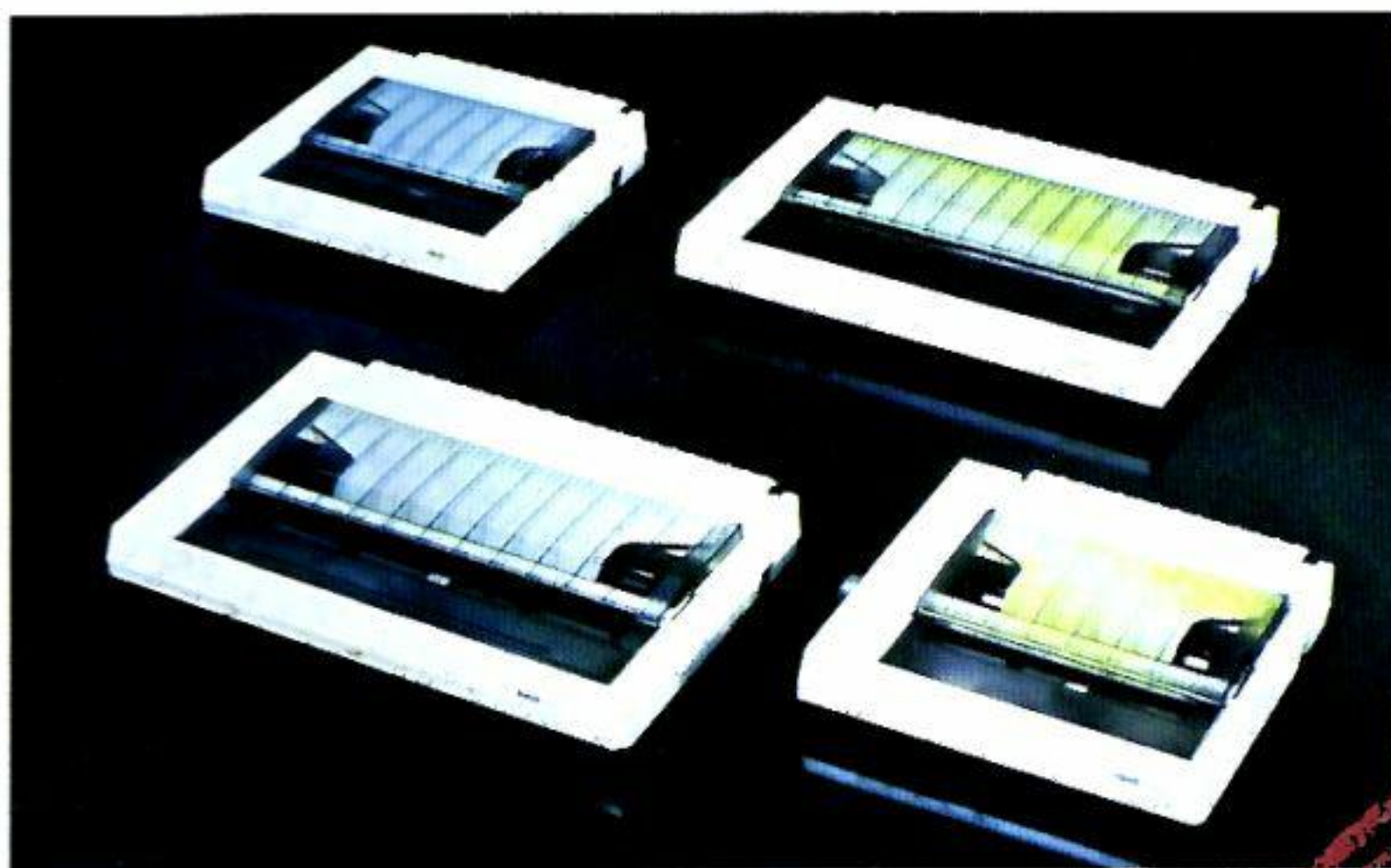
I lavori migliori verranno pubblicati su "Directory".

Prima di inviare il nastro (o disco) è consigliabile, però, dare un colpo di telefono in Redazione (Tel. 02/84.67.348) per stabilire se programma (e file-gioco) sono idonei alla pubblicazione.

TELCOM IMPORTA E DISTRIBUISCE LE STAMPANTI CITIZEN

Una gamma di stampanti che copre tutte le esigenze:
- stampa alfanumerica normale, ascendente-discendente, espansa, compressa, Elite, Near Letter Quality, grafica compatibile IBM, 8 Kbytes di buffer di stampa, trattore di moduli continui, inseritore di fogli singoli, testina a 9 aghi (24 aghi su HQP 45).

MODELLO	NR. COLONNE	VELOCITÀ (CAR/SEC)
LSP 10	80	120
MSP 10	80	160
MSP 15	136	160
MSP 20	80	200
MSP 25	136	200
HQP 45	136	200

E per gli hobbisti... la piccola 120D: 80 colonne, 120 caratteri car/sec con interfaccia parallela, seriale, compatibile **APPLE, COMMODORE.**

UNA ALTERNATIVA DA TENERE PRESENTE!

Senza alcun impegno desidero ricevere:
☐ Documentazione tecnica relativa a:

☐ Visita di un Vs. Funzionario

Nome e Cognome

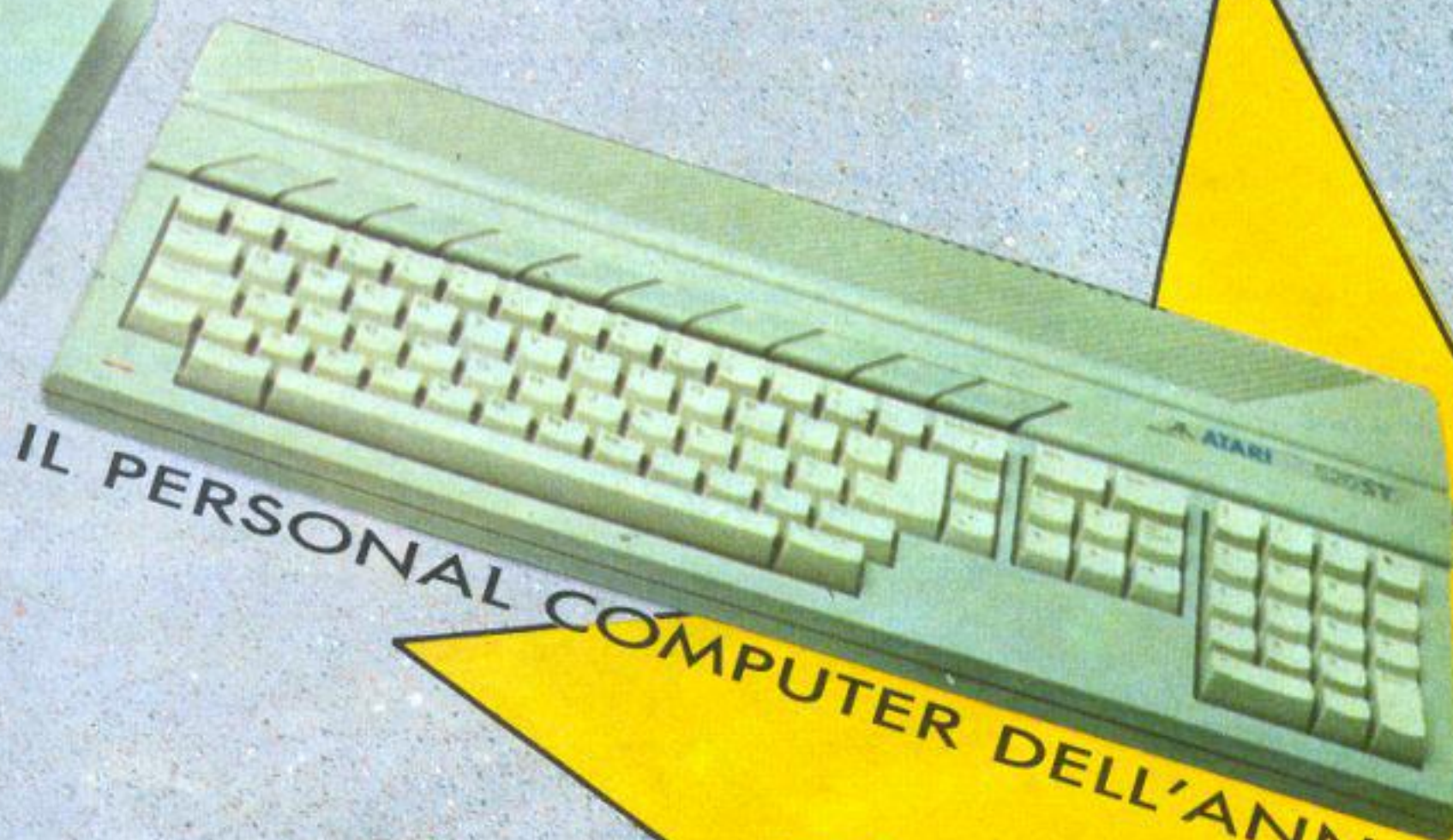
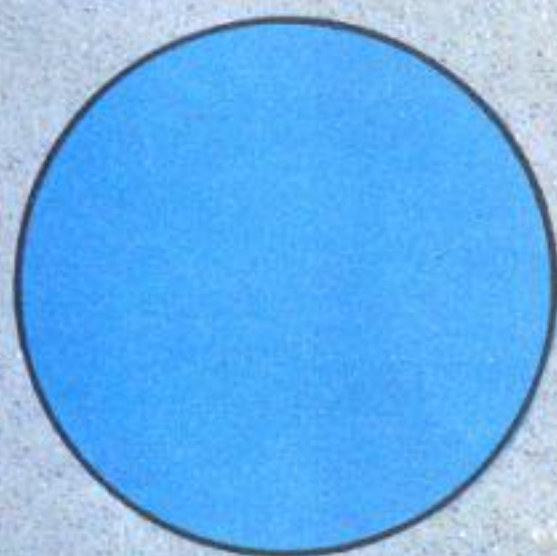
Società/Ente

Via

Città CAP

Telefono

CCC

Telcom srl - 20148 Milano - Via M. Civitali, 75
Tel. 02/4047648-4049046
Telex 335654 TELCOM I - Telefax 02/437964

anch'essa scomposta nei suoi dati fondamentali NO$ e AT$.

Come prima operazione la routine scorre tutte le posizioni A$(0,X) della matrice alla ricerca di dati attinenti l'espressione soggetto. Se non ne trova emette il verdetto "Nessun dato per confermare".

In caso affermativo, invece, scorre tutte le posizioni della colonna X in cui ha trovato l'espressione soggetto richiesta, incrementando Y da 1 fino a che trova una posizione vuota oppure una stringa uguale alla espressione attributo richiesta in input.

Se la trova dichiara "Posso confermare" e rinvia direttamente il controllo al "Supervisore"; se non la trova il programma non si arrende, ma mette il funzione la sua capacità di deduzione su tutti i dati disponibili. Ciò allo scopo di vedere se i dati permetton di dedurre ugualmente una conferma logicamente fondata.

Allo scopo scorre tutte le posizioni in cui siano contenuti attributi, anche relativi ad altre espressioni soggetto. Solo se non trova nell'intera matrice l'attributo uguale a quello richiesto in input, si arrende e afferma di non poter confermare.

Se invece trova un attributo uguale a quello richiesto in input in una colonna capeggiata da una espressione soggetto diversa da quella in input, allora verifica se tra gli attributi del soggetto in input ve ne sia uno che è uguale al capo-colonna che possiede l'attributo in questione.

Insomma deduce che un attributo appartenente ad un capo-colonna può essere attribuito anche ad una espressione soggetto che abbia quel capo-colonna come attributo. Se, ad esempio:

il falco è un uccello

...e se...

un uccello è un pennuto

...allora si può dedurre che:

il falco è un pennuto

Fatta la deduzione viene inviato il messaggio "Posso confermare", e aggiunto il nuovo attributo anche alla colonna capeggiata dal soggetto contenuto nell'input dell'utente, sempre che vi siano posizioni disponibili non occupate.

L'ultima routine del programma ("Visualizzazioni") offre all'utente una sorta di Utility in grado di visualizzare in ogni momento quali attributi siano già stati attribuiti ad un certo soggetto.

Considerazioni finali

Naturalmente il programma rivela i suoi limiti nella laboriosità della operazione di deduzione e anche in una certa semplificazione delle operazioni logiche eseguibili. Tuttavia si può ricavarne una impressione favorevole se consideriamo che la quantità dei dati che si possono immettere può essere anche molto grande (basta infatti espandere il dimensionamento della matrice fino ai limiti della memoria del computer) e che la operazione di deduzione e di memorizzazione degli attributi dedotti espande di molto i dati immessi direttamente.

Un altro limite risiede nella rigidità dell'interfaccia con l'utente che è costretto ad immettere frasi fortemente limitate nella loro sintassi a causa della povertà del trattamento del linguaggio naturale che il programma è in grado di compiere. Tuttavia già la routine che elimina dall'input gli articoli testimonia che è possibile operare efficacemente sul linguaggio naturale. Si ricorda che le espressioni soggetto e attributo non sono declinabili nè per genere nè per numero, la qual cosa può generare problemi rilevanti.

```
10 REM OPERATORE LOGICO
11 :
12 REM ESEMPIO DI INTELLIGENZA
   ARTIFICIALE BY SANDRO CERT
   I & FRANCO TOLDI
13 :
14 REM PER QUALSIASI COMMODORE
15 :
20 REM ROUTINE SUPERVISORE
30 GOSUB 100:REM INIZIALIZZAZI
   ONE
40 WW=0:GOSUB 220:IF WW THEN 4
   0:REM RICEVE INPUT
50 GOSUB 280:REM  SMISTA E ELA
   BORA INPUT
60 IF P=1 THEN GOSUB 500:GOTO
   40:REM DICHIARATIVE
70 IF P=2 THEN GOSUB 610:GOTO
   40:REM INTERROGATIVE
80 IF P=3 THEN GOSUB 840:REM
   VISUALIZZA LISTE
90 GOTO 40
100 REM INIZIALIZZAZIONE
110 DIM A$(19,19)
120 FOR Y=0 TO 19
130 FOR X=0 TO 19
140 A$(Y,X)="@":REM CARATTERE "
    SPIA"
150 NEXTX:NEXTY:PRINTCHR$(147):
    GOSUB 1000:RETURN
220 REM RICEVE INPUT
230 P=0
250 INPUT "(HELP D- I- V-)";I$:
    IF I$="HELP" THEN GOSUB 100
    0:RETURN
260 IF I$="FINE" THEN END
270 RETURN
280 REM  SMISTA
290 B$=LEFT$(I$,2)
300 I$=MID$(I$,3)
```

```
310 IF B$="D-" THEN P=1:GOTO 35
    0
320 IF B$="I-" THEN P=2:GOTO 35
    0
330 IF B$="V-" THEN P=3:GOTO 35
    0
340 PRINT"INPUT NON CONFORME AL
    LE REGOLE":GOTO 490
350 REM DIVIDE INPUT
360 IF P=3 THEN NO$=I$:GOTO 410
370 FOR K=1 TO LEN(I$)
380 IF MID$(I$,K,4)=" E' " THEN
     NO$=LEFT$(I$,K-1):AT$=MID$
    (I$,K+4):GOTO 410
390 NEXT
400 PRINT"INPUT NON CONFORME AL
    LE REGOLE":GOTO 490
410 REM TOGLIE ARTICOLI
420 RESTORE
430 FOR XX=1 TO 8
440 READ AR$
450 IF MID$(NO$,1,LEN(AR$))=AR$
     THEN NO$=MID$(NO$,LEN(AR$)
    +1)
460 IF MID$(AT$,1,LEN(AR$))=AR$
     THEN AT$=MID$(AT$,LEN(AR$)
    +1)
470 NEXT
480 DATA "IL ","LO ","LA ","L'"
    ,"UN ","UNA ","UN'","UNO "
490 RETURN
500 REM STORAGE FRASI DICHIARAT
    IVE
510 FOR X=0 TO 19
520 IF A$(0,X)=NO$ THEN 550
530 IF A$(0,X)="@" THEN A$(0,X)
    =NO$:GOTO 550
540 NEXT
550 FOR Y=1 TO 19
560 IF A$(Y,X)=AT$ THEN 590
570 IF A$(Y,X)="@" THEN A$(Y,X)
    =AT$:GOTO 590
580 NEXT
590 PRINT"OK"
600 RETURN
610 REM INTERROGATIVE
620 FOR X=0 TO 19
630 IF A$(0,X)=NO$ THEN 660
640 NEXT
650 PRINT"NESSUN DATO PER CONFE
    RMARE":GOTO 830
660 FOR Y=1 TO 19
670 IF A$(Y,X)=AT$ THEN 820
680 NEXT
690 FOR N=1 TO 19
700 FOR M=0 TO 19
710 IF A$(N,M)=AT$ THEN 740
720 NEXT:NEXT
730 GOTO 770
740 FOR Y=1 TO 19
750 IF A$(Y,X)=A$(0,M) THEN 780
760 NEXT
770 PRINT"NESSUN DATO PER CONFE
    RMARE":GOTO 830
780 FOR S=Y+1 TO 19
790 IF A$(S,X)="@" THEN A$(S,X)
    =AT$:GOTO 820
800 NEXT
810 PRINT"NON HO PIU' SPAZIO":G
    OTO 830
820 PRINT"POSSO CONFERMARE "
830 RETURN
840 REM VISUALIZZAZIONI
850 FOR K=0 TO 19
860 IF A$(0,K)=NO$ THEN 900
870 NEXT
880 PRINT"NESSUN DATO DISPONIBI
    LE"
890 GOTO 940
900 PRINTNO$+" E' :"
910 FOR J=1 TO 19
920 IF A$(J,K)<>"@" THEN PRINTA
    $(J,K)
930 NEXT
940 RETURN
999 REM MESSAGGIO DI HELP
1000 PRINT:PRINT
1010 PRINT"PER LE DICHIARATIVE F
     AR PRECEDERE D-"
1020 PRINT"PER LE INTERROGATIVE
     FAR PRECEDERE I-"
1030 PRINT"PER LE VISUALIZZAZION
     I FAR PRECEDERE V-":PRINT
1035 PRINT"[RVS]HELP[RVOFF] PER
     RICHIAMARE REGOLE":PRINT
1040 PRINT"[RVS]FINE[RVOFF] PER
     FINIRE"
1050 PRINT:PRINT"PREMI UN TASTO"
1060 GET WW$:IF WW$="" THEN 1060
1070 WW=1:RETURN
```

Corsa all'ippodromo

Una simulazione ben curata nell'animazione di tre cavalli spinti al galoppo sfrenato

di Carlo & Lorenzo Barazzetta

Questo programma, scritto per divertire coloro che possiedono il C/64 o il C/128 (da usare, ovviamente, in "modo" 64), sarà utilissimo anche per chi intende studiare particolari tecniche di programmazione in LM.

Sfortunatamente per gli altri computer della famiglia Commodore, solamente il C/64 è dotato di sprite che permettono in modo semplicissimo di "animare" figure e di creare, con brevi listati, videogiochi di sicuro effetto.

In questo caso si tratta di una corsa fra tre cavalli guidati da altrettanti concorrenti i quali devono frustare (perdonateci, amici del WWF) il proprio animale per vincere il derby.

Fate attenzione, però, che il cavallo, giustamente seccato da eventuali eccessi, rallenterà la corsa per favi dispetto e sarà difficile riguadagnare terreno.

Alla partenza i tre cavalli si dirigeranno verso la parte destra dello schermo e durante la corsa quelli più lenti rallenteranno, restando a sinistra. Inoltre uno SCROLL (movimento) del video (realizzato pixel per pixel in LM) aumenterà il realismo del gioco e lo scalpitio degli zoccoli renderà l'atmosfera più avvincente.

Per spronare i cavalli, usate i tasti:

[DEL] per il cavallo in alto
[D] per il cavallo al centro
[M] per il cavallo in basso

Ricordiamo, inoltre, che il gioco è molto veloce grazie ad una routine in Linguaggio Macchina. Mentre questa viene eseguita (dall'inizio alla fine della corsa) non è possibile bloccare il gioco, ma bisognerà attendere l'arrivo affinchè i tasti Run/Stop e Restore funzionino nuovamente.

Dopo aver digitato il programma, salvatelo su nastro o disco prima di dare il Run.

Volendo potete modificare i parametri specificati con le REM, per cambiare colori e velocità (se preferite prati viola o cavalli azzurri fate pure: siamo in un Paese libero).

Se volete arricchire il gioco vi consigliamo di aggiungere (se ci riuscite) la gestione delle scommesse sulla corsa in modo da giocare in più persone: tre giocatori, ad esempio, prenderannpo le vesti dei fantini e gli altri i bookmakers!

Terminata la gara potete, ad esempio, controllare che la coordinata X dei cavalli sia uguale a 255 (la corsa può anche terminare con più vincitori) mediante un'istruzione PEEK relativa alle tre locazioni: 53248 per il cavallo in alto, 53250 per quello al centro e 53252 per quello in basso. Potete anche aggiungere la richiesta dei nomi dei fantini e dei rispettivi cavalli, stabilire una serie di gare con punteggio e classifica finale...

Se invece siete pigri, limitatevi a giocare, ma fate almeno lo sforzo di invitare alcuni amici: da soli è troppo facile vincere!

Lo scrolling laterale

Qualche cenno sulla tecnica di Scroll utilizzata nel gioco:

I bit 0, 1 e 2 della locazione $D016 controllano la posizione orizzontale dello schermo. Tre bit, come è noto, possono assumere un valore da 0 a 7 che indica, in questo caso, di quanti pixel i caratteri devono essere spostati verso destra.

Se i tre bit vengono modificati continuamente in modo tale che assumano i valori da 7 a 0, i caratteri dello schermo si sposteranno da destra verso sinistra di un pixel alla volta.

Quando ciò avviene i caratteri delle staccionate danno l'impressione di spostarsi continuamente verso sinistra anche se, in realtà, non fanno altro che spostarsi di sette pixel verso sinistra per poi ritornare nella posizione iniziale.

Ciò è possibile perchè ogni carattere dello schermo ne ha uno identico a se stesso sia a destra che a sinistra.

Lo spostamento non influisce però sugli sprite che restano fissi nella propria posizione.

Prima di effettuare lo scroll occorre azzerare il bit 3 della locazione $D016 che riduce lo schermo da 40 a 38 colonne. In questo modo, infatti, si evitano sgradevoli effetti ai bordi.

```
100 REM  COMMODORE 64
105 REM  LA CORSA DEI CAVALLI
110 REM  BASIC+LM
115 REM  BY CARLO & LORENZO BAR
    AZZETTA
120 :
125 DA=16256:VI=53248:SI=54272
130 FOR A=0 TO 2:POKE SI+A*7+4,
    8:NEXT
135 A=8:REM  VOLUME
140 POKE SI+24,A OR 128
145 POKE SI+14,255:POKE SI+15,2
    55
150 POKE SI+18,129
155 POKE SI,0:POKE SI+1,10
160 POKE SI+5,1:POKE SI+6,0
165 PRINTCHR$(147)CHR$(8)
170 SYS58792:REM  INIZIALIZZAZI
    ONE VIDEO
175 POKE VI+32,0:REM  COLORE BO
    RDO
180 POKE VI+33,5:REM  COLORE SF
    ONDO
185 POKE 646,PEEK(VI+32) AND 15
190 PRINT"LA CORSA DEI CAVALLI"
195 PRINT:PRINT:PRINT
200 PRINT"PER FRUSTARE I CAVALL
    I";
205 PRINTCHR$(32)"USARE I TASTI
    :":PRINT
210 PRINT"[DEL] CAVALLO ALTO":P
    RINT
215 PRINT"[ D ] CAVALLO CENTRO"
    :PRINT
220 PRINT"[ M ] CAVALLO BASSO":
    PRINT
225 PRINT:PRINT:PRINT"ATTENDERE
    "
230 :
235 REM  INIZIO DATI LM
240 :
245 DATA  120,216,173,13,64,201
    ,33,208
250 DATA  14,173,14,64,208,9,17
    3,22,208
255 DATA  201,199,208,2,88,96,1
    62,2,254
260 DATA  0,64,240,5,202,16,248
    ,48,223
265 DATA  189,3,64,157,0,64,222
    ,6,64,208
270 DATA  104,169,16,157,6,64,1
    89,248
275 DATA  7,73,1,157,248,7,160,
    128,140
280 DATA  4,212,200,140,4,212,1
    38,168
285 DATA  169,127,56,42,42,136,
    16,251
290 DATA  141,0,220,73,255,44,1
    ,220,208
295 DATA  24,13,9,64,141,9,64,1
    89,3,64
300 DATA  205,10,64,240,48,14,2
    7,212,233
305 DATA  0,157,3,64,176,38,44,
    9,64,8
310 DATA  73,255,45,9,64,141,9,
    64,40,240
315 DATA  223,173,27,212,41,7,1
    25,3,64
320 DATA  176,7,205,11,64,144,5
    ,240,3
325 DATA  173,10,64,157,3,64,13
    8,10,168
330 DATA  185,0,208,105,1,240,5
    ,153,0
335 DATA  208,208,78,160,4,152,
    74,141
340 DATA  12,64,236,12,64,240,1
    1,185,0
345 DATA  208,240,6,56,233,1,15
    3,0,208
350 DATA  136,136,16,231,169,7,
    44,22,208
355 DATA  240,5,206,22,208,48,3
    9,172,13
360 DATA  64,192,40,176,26,169,
    160,153
```

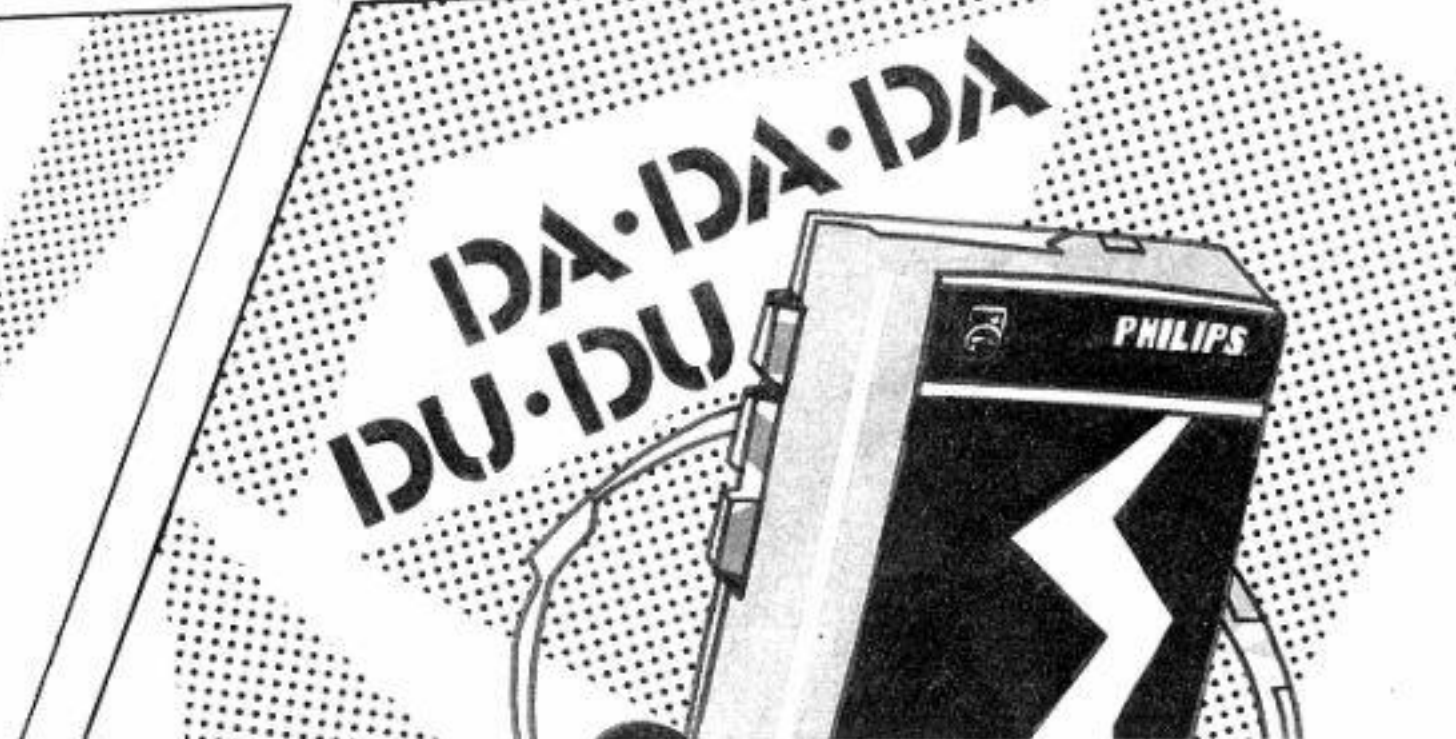
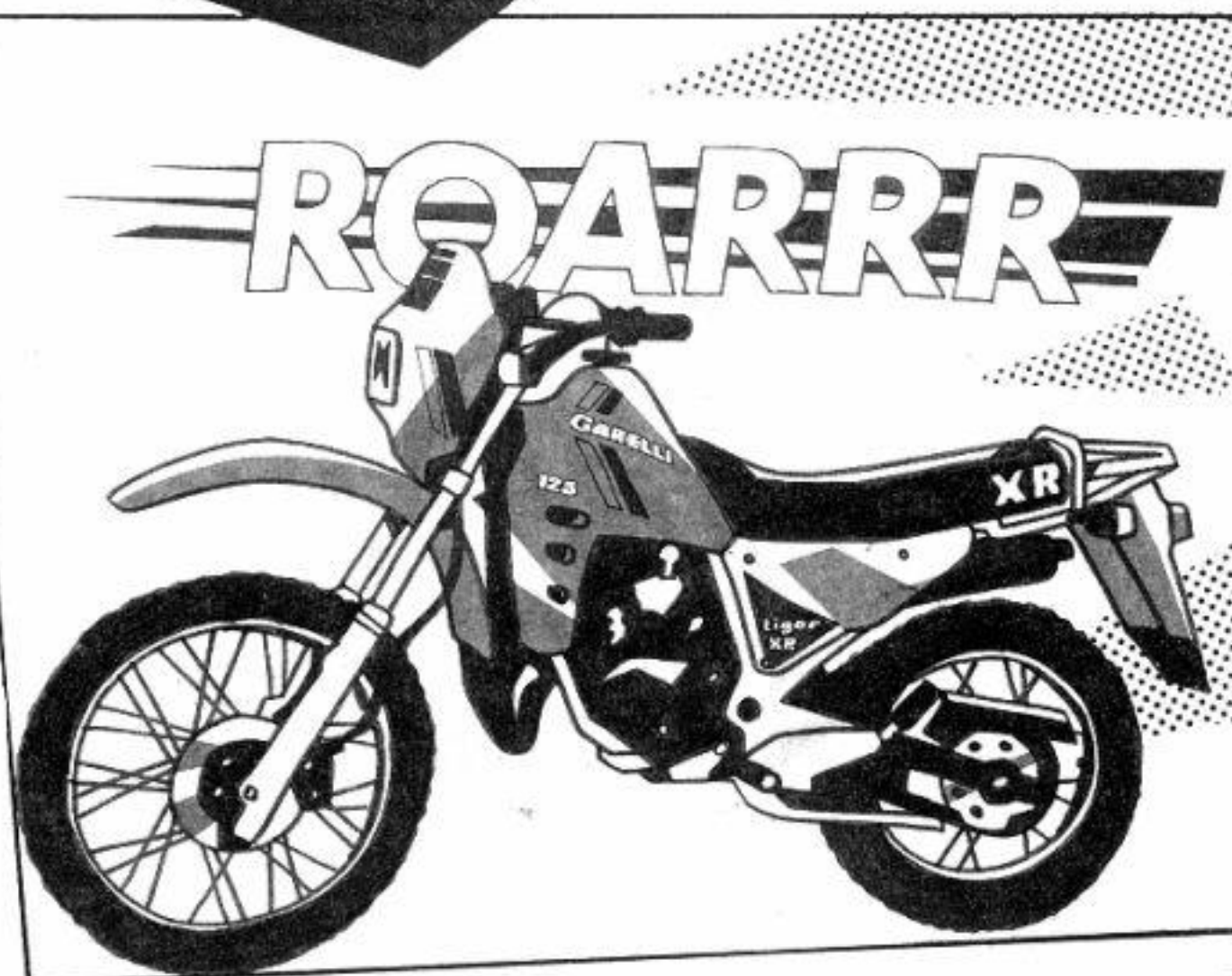
```
365 DATA  0,4,169,7,136,48,13,1
     41,22,208
370 DATA  173,14,64,9,176,153,0
     ,4,208
375 DATA  6,206,14,64,141,22,20
     8,206,13
380 DATA  64,76,46,64
385 :
390 REM  FINE DATI LM
395 :
400 FOR A=0 TO 245
405 PRINTCHR$(145)SPC(9)373-ACH
     R$(20)
410 READ B:IF B>=0 AND B<256 TH
     EN 420
415 PRINT"ERRORE NEI DATI LM";:
     END
420 POKE DA+144+A,B:C=C+B:NEXT
425 IF C<>25186 THEN 415
430 :
435 REM  INIZIO DATI SPRITE
440 :
445 DATA  0,0,0,0,0,0,0,0,0,0,0
     ,0,0,12
450 DATA  0,0,15,0,0,28,0,0,62,
     16,0,56
455 DATA  96,0,63,240,119,231,2
     40,255
460 DATA  231,184,143,207,24,15
     ,223,128
465 DATA  15,255,128,11,129,192
     ,5,129
470 DATA  96,6,226,192,1,53,128
     ,1,179
475 DATA  0,0,0,0,0,0,0,0,0,0,0
     ,0,0,0
480 DATA  0,0,0,0,24,0,0,30,0,0
     ,56,16
485 DATA  0,63,96,0,113,240,0,1
     23,248
490 DATA  183,231,156,255,199,1
     2,111,159
495 DATA  0,15,255,128,7,255,22
     4,15,131
500 DATA  176,62,0,216,108,0,10
     8,216,0
505 DATA  102,48,0,6,0,0,0,0,-1
510 :
515 REM  FINE DATI SPRITE
520 :
525 FOR A=0 TO 127
530 PRINTCHR$(145)SPC(9)127-ACH
     R$(20)
535 READ B:IF B>=0 AND B<256 TH
     EN 545
540 PRINT"ERRORE NEI DATI SPRIT
     E";:END
545 POKE DA+A,B:C=C+B:NEXT
550 IF C<>33907 THEN 540
555 :
560 PRINTCHR$(147):POKE VI+22,7
565 C(0)=7:REM  COLORE STACCION
     ATA 1
570 C(1)=6:REM  COLORE STACCION
     ATA 2
575 C(2)=0:REM  COLORE STACCION
     ATA 3
580 FOR A=0 TO 2:POKE VI+A*2+1,
     A*56+70
585 POKE 646,C(A):FOR B=1 TO 5:
     PRINT:NEXT
590 FOR B=1 TO 40:PRINTCHR$(178
     );:NEXT
595 FOR B=1 TO 40:PRINTCHR$(125
     );:NEXTB,A
600 POKE VI+39,0:REM  COLORE CA
     VALLO 1
605 POKE VI+40,1:REM  COLORE CA
     VALLO 2
610 POKE VI+41,9:REM  COLORE CA
     VALLO 3
615 POKE VI+27,7:POKE VI+23,7:P
     OKE VI+29,7
620 POKE DA+138,80:REM  VELOCIT
     A' MIN
625 POKE DA+139,180:REM  VELOCI
     TA' MAX
630 POKE 646,PEEK(VI+32) AND 15
635 PRINTCHR$(18);
640 :
645 PRINTCHR$(19);
650 FOR A=1 TO 40:PRINTCHR$(32)
     ;:NEXT
655 FOR A=0 TO 2:POKE 2040+A,25
     5
660 B=(PEEK(DA+138)+PEEK(DA+139
     ))/2
665 POKE DA+128+A,B:POKE DA+131
     +A,B
670 POKE DA+134+A,16:NEXT
```

```
675 POKE DA+137,0
680 POKE DA+141,255:POKE DA+142
    ,9
685 POKE VI+21,7:SYSDA+144:POKE
     SI+4,8
690 C=7:FOR A=0 TO 2
695 IF PEEK(VI+A*2)=255 THEN C=
    C-2↑A
700 NEXT
705 FOR A=1 TO 40:FOR B=1 TO 10
    0:NEXT
710 POKE VI+21, NOT PEEK(VI+21)
     AND 7 OR C
715 NEXT
720 PRINTCHR$(19)SPC(9);
725 PRINT"SPAZIO PER RIPARTIRE"
    ;
730 A=60:REM  TASTO PER RIPARTI
    RE
735 IF PEEK(203)<>A THEN 735
740 POKE VI+21,0
745 FOR A=0 TO 2:POKE VI+A*2,0:
    NEXT
750 GOTO 645
755 :
760 REM  FINE LISTATO
```

```
Disassemblato commentato del gioco "Corsa all'ippodromo"

Locazioni di memoria utilizzate nel programma

$4000,$4001,$4002 locazioni utilizzate per la velocita'
                  dei cavalli
$4003,$4004,$4005 valori di velocita' dei cavalli
$4006,$4007,$4008 locazioni utilizzate per il ritardo del
                  movimento dalle zampe
$4009             maschera pressione tasti
$400a             valore di velocita' minima
$400b             valore di velocita' massima
$400c             valore transitorio per cpx
$400d             byte basso distanza dall'arrivo
$400e             byte alto distanza dall'arrivo
$400f             locazione non utilizzata


4010 78       sei          ;istruzioni preliminari
4011 d8       cld          ;

4012 ad 0d 40 lda $400d    ;test sulla fine della corsa
4015 c9 21    cmp #$21     ;
4017 d0 0e    bne $4027    ;
4019 ad 0e 40 lda $400e    ;
401c d0 09    bne $4027    ;
401e ad 16 d0 lda $d016    ;
4021 c9 c7    cmp #$c7     ;
4023 d0 02    bne $4027    ;

4025 58       cli          ;torna al programma Basic
4026 60       rts          ;

4027 a2 02    ldx #$02     ;ciclo di ritardo per il movimento
4029 fe 00 40 inc $4000,x;dei cavalli
402c f0 05    beq $4033    ;
402e ca       dex          ;
402f 10 f8    bpl $4029    ;
4031 30 df    bmi $4012    ;

4033 bd 03 40 lda $4003,x;ripristina il valore del ritardo
4036 9d 00 40 sta $4000,x;

4039 de 06 40 dec $4006,x;ritardo movimentc zampe dei
403c d0 68    bne $40a6    ;cavalli

403e a9 10    lda #$10     ;ripristina il valore del ritardo
4040 9d 06 40 sta $4006,x;

4043 bd f8 07 lda $07f8,x;movimento zampe dei cavalli
4046 49 01    eor #$01     ;
4048 9d f8 07 sta $07f8,x;

104b a0 80    ldy #$80     ;rumore zoccoli dei cavalli
404d 8c 04 d4 sty $d404    ;
4050 c8       iny          ;
4051 8c 04 d4 sty $d404    ;

4054 8a       txa          ;scansione tastiera:
4055 a8       tay          ;controlla la pressione del tasto
4056 a9 7f    lda #$7f     ;corrispondente al cavallo
4058 38       sec          ;
4059 2a       rol          ;
405a 2a       rol          ;
405b 88       dey          ;
405c 10 fb    bpl $4059    ;
405e 8d 00 dc sta $dc00    ;
4061 49 ff    eor #$ff     ;
4063 2c 01 dc bit $dc01    ;
4066 d0 18    bne $4080    ;
4068 0d 09 40 ora $4009    ;se il tasto non e' premuto
406b 8d 09 40 sta $4009    ;diminuisce casualmente la
406e bd 03 40 lda $4003,x;velocita' del cavallo
4071 cd 0a 40 cmp $400a    ;
4074 f0 30    beq $40a6    ;
4076 0e 1b d4 asl $d41b    ;
4079 e9 00    sbc #$00     ;
407b 9d 03 40 sta $4003,x;
407e b0 26    bcs $40a6    ;

4080 2c 09 40 bit $4009    ;se il tasto e' premuto aumenta
4083 08       php          ;casualmente la velocita' del
4084 49 ff    eor #$ff     ;cavallo
4086 2d 09 40 and $4009    ;
4089 8d 09 40 sta $4009    ;
408c 28       plp          ;
408d f0 df    beq $406e    ;
408f ad 1b d4 lda $d41b    ;
4092 29 07    and #$07     ;
4094 7d 03 40 adc $4003,x;
4097 b0 07    bcs $40a0    ;
4099 cd 0b 40 cmp $400b    ;
409c 90 05    bcc $40a3    ;
409e f0 03    beq $40a3    ;
40a0 ad 0a 40 lda $400a    ;
40a3 9d 03 40 sta $4003,x;

40a6 8a       txa          ;sposta il cavallo verso destra,
40a7 0a       asl          ;se si trova nella posizione
40a8 a8       tay          ;massima sposta gli altri cavalli
40a9 b9 00 d0 lda $d000,y;e lo schermo (tramite scroll)
40ac 69 01    adc #$01     ;indietro
40ae f0 05    beq $40b5    ;
40b0 99 00 d0 sta $d000,y;
40b3 d0 4e    bne $4103    ;
40b5 a0 04    ldy #$04     ;
40b7 98       tya          ;
40b8 4a       lsr          ;
40b9 8d 0c 40 sta $400c    ;
40bc ec 0c 40 cpx $400c    ;
40bf f0 0b    beq $40cc    ;
40c1 b9 00 d0 lda $d000,y;
40c4 f0 06    beq $40cc    ;
40c6 38       sec          ;
40c7 e9 01    sbc #$01     ;
40c9 99 00 d0 sta $d000,y;
40cc 88       dey          ;
40cd 88       dey          ;
40ce 10 e7    bpl $40b7    ;

40d0 a9 07    lda #$07     ;scroll dello schermo
40d2 2c 16 d0 bit $d016    ;
40d5 f0 05    beq $40dc    ;
40d7 ce 16 d0 dec $d016    ;
40da 30 27    bmi $4103    ;
40dc ac 0d 40 ldy $400d    ;
40df c0 28    cpy #$28     ;
40e1 b0 1a    bcs $40fd    ;
40e3 a9 a0    lda #$a0     ;
40e5 99 00 04 sta $0400,y;
40e8 a9 07    lda #$07     ;
40ea 88       dey          ;
40eb 30 0d    bmi $40fa    ;
40ed 8d 16 d0 sta $d016    ;
40f0 ad 0e 40 lda $400e    ;
40f3 09 b0    ora #$b0     ;
40f5 99 00 04 sta $0400,y;
40f8 d0 06    bne $4100    ;
40fa ce 0e 40 dec $400e    ;
40fd 8d 16 d0 sta $d016    ;
4100 ce 0d 40 dec $400d    ;
4103 4c 2e 40 jmp $402e    ;
```

GRANDE CONCORSO ANTONELLI

Tutta un'altra musica!

AUT. MIN. N. 4/292624 DEL 30/06/1986

ANTONELLI CAMBIA MUSICA
Intendiamoci, la musica dovete farla voi, ma con Antonelli é facile imparare. E poi, con il concorso, avete uno stimolo in più.

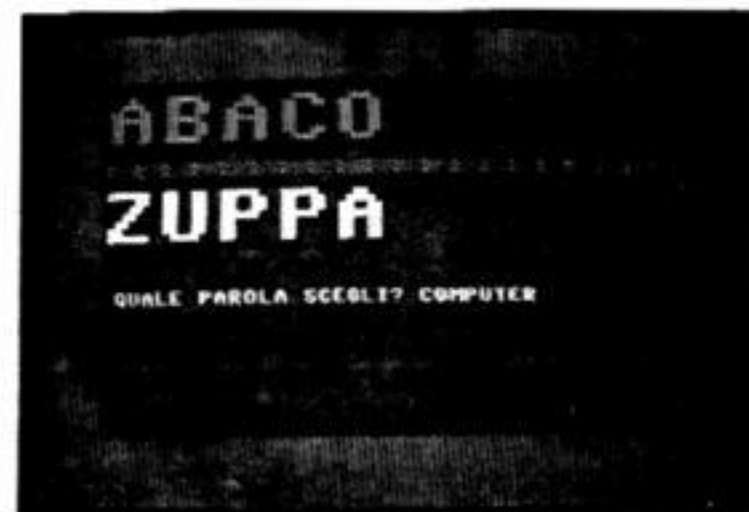
COSA SI VINCE?
Dunque, 50 Riproduttori in cuffia stereo (Philips), 6 impianti HI-FI stereo, 3 Moto Garelli "TIGER 125 XR" e perfino una Fuoristrada ARO 10.1 Super Ischia 4x4

E COME SI FA?
Per partecipare al concorso occorre acquistare un organo Antonelli e spedire la garanzia entro il 10/1/1987, dopo averla fatta timbrare dal rivenditore. Poi, finché aspettate i risultati delle estrazioni, potete sperimentare le mille possibilità degli organi Antonelli per fare musica, da soli o con gli amici.

tel. 071/7100184

Cercaparola

Un gioco di società per divertirsi con gli amici e trascorrere il tempo piacevolmente

di Roberto Morassi

Il passatempo "Cercaparola" è ispirato ad un divertente gioco proposto di recente in TV.

Può essere giocato da un numero qualunque di giocatori, uno dei quali (chiamiamolo "arbitro") introduce segretamente nel computer una parola che gli altri, a turno, devono cercare di indovinare.

Supponiamo che la parola segreta sia PANE. Vengono date, all'inizio, le due parole estreme del vocabolario (per semplicità: ABACO e ZUPPO; cfr. ultime righe del programma) che delimitano il campo nel quale è compresa la parola da indovinare.

Un primo tentativo, ad esempio GATTO, ridefinirà uno dei limiti della ricerca, che proseguirà fra GATTO e ZUPPO; se si propone adesso TOPO, i nuovi limiti saranno GATTO e TOPO, e così via. Poichè il campo di ricerca diventa sempre più stretto, alla fine qualcuno riuscirà a identificare la parola nascosta, e si può ripartire nuovamente con un nuovo "arbitro".

Come funziona il programma

Il programma stampa le due parole-limite in formato gigante. A tale scopo vengono costruite delle "repliche" (in proporzione 16:1) dei caratteri standard, usando una matrice di stringhe grafiche a media risoluzione.

La costruzione di questa matrice (righe 410-440) è piuttosto complicata, e richiede circa 30 secondi durante i quali la tastiera è disabilitata (è inutile, pertanto, premere Run/Stop e Restore).

Viene poi richiesto di digitare "alla cieca" (dal momento che il colore della stampa viene posto uguale a quello del fondo) la parola segreta, che dovrà avere una lunghezza massima di dieci lettere.

Questa limitazione, che vale per tutto il gioco, è dovuta solamente a motivi estetici: i caratteri giganti, infatti, sono larghi quattro volte quelli normali, e una parola di dieci lettere occupa un'intera riga di schermo.

Dopo la visualizzazione delle prime due parole-limite, iniziano i tentativi di indovinare quella nascosta. Le parole via via proposte vengono accettate solo se non superiori alle dieci lettere, e se comprese entro il campo di ricerca (pena un opportuno messaggio di errore).

Dopo ogni INPUT vengono richiamate due brevi routine in linguaggio macchina, la prima delle quali (SYS 890) riempie lo schermo di lettere casuali e continuamente variabili, mentre la seconda (SYS 820,N, dove N èun codice-colore) pulisce lo schermo, sempre con metodo casuale, con spazi-reverse di colore N.

Le due routine, caricate nel buffer del registratore con i DATA delle linee 200-360, sono interamente rilocabili.

Un'altra micro-routine (SYS 950,N) cancella la riga N-ma in caso di INPUT illegale.
Le linee 580-600, infine, provvedono ad eseguire gli opportuni confronti di stringhe e a ridefinire le due parole-limite: il ciclo viene ripetuto finchè qualcuno non indovina la parola misteriosa.

Naturalmente si può ampliare il programma, aggiungendovi, ad esempio, effetti sonori, una routine per stabilire i turni, un sistema di punteggi, e così via.

```
 10 REM CERCAPAROLA
 12 :
 15 REM GIOCO PER C/64
 18 :
 20 REM BY ROBERTO MORASSI
 30 REM PISTOIA
120 :
130 POKE 53281,7:PRINTCHR$(147)
     TAB(255)SPC(230)"ATTENDERE
     35 SECONDI"
140 DIM X,Y,C$,A,B,W,Z,N,P$,A$,
     B$,S$,CO$
150 DIM C$(3,3),L$(25,4)
160 DEF  FNC(X)=(((X AND (2*N))
     +(X AND N))/N)
165 FOR X=1 TO 32:READ Y:C$=C$+
     CHR$(Y):NEXT
180 FOR X=0 TO 3:FOR Y=0 TO 3:C
     $(X,Y)=MID$(C$,8*X+2*Y+1,2)
     :NEXT:NEXT
190 :
191 DATA  146,32,146,172,146,18
     7,146,162,146,188,18,161,18
     ,191,18,190,146
192 DATA  190,146,191,146,161,1
     8,188,18,162,18,187,18,172,
```

```
    18,32
193 :
200 DATA  032,241,183,169,255,1
    41,015,212
210 DATA  169,128,141,018,212,1
    69,000,168
220 DATA  032,219,255,173,027,2
    12,133,251
230 DATA  173,027,212,041,003,0
    09,004,133
240 DATA  252,169,160,145,251,1
    65,252,024
250 DATA  105,212,133,252,138,1
    45,251,234
260 DATA  234,234,234,234,234,2
    34,234,036
270 DATA  162,016,216,140,015,2
    12,140,018
280 DATA  212,169,147,076,210,2
    55,169,255
290 DATA  141,015,212,169,128,1
    41,018,212
300 DATA  169,000,168,032,219,2
    55,173,027
310 DATA  212,133,251,069,162,0
    41,031,170
320 DATA  173,027,212,041,003,0
    09,004,133
330 DATA  252,138,145,251,234,2
    34,234,234
340 DATA  234,234,234,234,234,2
    34,036,162
350 DATA  080,220,140,015,212,1
    40,018,212
360 DATA  096,000,032,241,183,0
    76,255,233
370 :
380 FOR X=0 TO 135:READ Y:POKE
    820+X,Y:A=A+Y: NEXT
390 IF A<>20358 THEN PRINT"ERRO
    RE NEI DATA":STOP
400 :
410 POKE 56334,PEEK(56334) AND
    254:POKE 1,PEEK(1) AND 251
420 FOR X=0 TO 25:FOR Y=1 TO 4:
    A=53256+8*X+2*(Y-1):W=PEEK(
    A):Z=PEEK(A+1)
430 FOR B=7 TO 1 STEP -2:N=2↑(B
    -1):L$(X,Y)=L$(X,Y)+C$(FNC(
    W),FNC(Z))
435 NEXT:NEXT:NEXT
440 POKE 1,PEEK(1) OR 4:POKE 56
    334,PEEK(56334) OR 1
450 :
460 POKE 53281,6:PRINTCHR$(147)
    "         INSERISCI LA PAROLA
    SEGRETA"
470 PRINTSPC(7)"(NON PIU' DI 10
     LETTERE !)
480 PRINTSPC(10)"POI PREMI [RET
    URN][BLEU]":INPUT P$
490 IF LEN(P$)>10 THEN 460
500 A$="ABACO":B$="ZUPPO"
510 PRINT"[BIANCO]"CHR$(147) TA
    B(40);:CO$="[VERDE]":C$=A$:
    GOSUB 650
520 PRINT TAB(40)"████████████
    ████████████████████████
530 PRINT TAB(40);:CO$="[GIALLO
    ]":C$=B$:GOSUB 650
540 PRINT"[3 DOWN]";
550 INPUT "[BIANCO] QUALE PAROL
    A SCEGLI";S$
560 IF LEN(S$)>10 THEN SYS950,1
    5:PRINT"[UP]";:GOTO 550
570 IF S$<=A$ OR S$>=B$ THEN PR
    INT"[2 DOWN]   NON VALIDA !
    !":FOR X=1 TO 500:NEXT:GOTO
     510
580 SYS890:FOR X=1 TO 1000:NEXT
    :IF S$=P$ THEN SYS820,2:GOT
    O 610
590 SYS820,6:IF S$<P$ THEN A$=S
    $:GOTO 510
600 B$=S$:GOTO 510
610 PRINT"INDOVINATO !!!! LA PA
    ROLA E' PROPRIO
620 PRINT TAB(160);:CO$="[NERO]
    ":C$=P$:GOSUB 650:PRINT:PRI
    NT
630 PRINT"[BIANCO]     ANCORA ?
    (S/N)":WAIT 198,1:GET A$:IF
     A$="S" THEN 460
640 END
650 PRINTCO$;:FOR X=1 TO 4:FOR
    Y=1 TO LEN(C$):PRINTL$(ASC(
    MID$(C$,Y))-65,X);:NEXT
660 IF LEN(C$)<10 THEN PRINT
670 NEXT:RETURN
680 :
```

Quanto vale la tua squadra del cuore?

Un metodo obiettivo (!) e valido per valutare le squadre in concorso, evitando alterazioni di giudizio derivanti dal tifo di parte

di Alessandro de Simone

Chi compila la schedina Totocalcio appartiene, inevitabilmente, ad una delle due categorie possibili: tifosi e non tifosi.

Nel primo caso il pronostico è falsato dall'opinione strettamente personale che si ha di ciascuna squadra;

nel secondo, al contrario, si rischia di compilare le colonne con una basilare incompetenza che porta, ovviamente, a limitare le probabilità di vincita.

Il giocatore ideale è quella persona che, pur avendo notevole competenza nel formulare giudizi su tutte le squadre di calcio, si rifiuta di parteggiare per una di esse in particolare: il che costituisce un assurdo. Non è ancora nato il tifoso impassibile nè l'esperto senza tifo.

Nel compilare una schedina, ad esempio, provate a chiedere il parere di un interista e di uno juventino: difficilmente la discussione che verrà fuori si terrà nei limiti di un pacato scambio di vedute.

Scherzi a parte, dovrà pur esserci un modo di individuare parametri obiettivi di valutazione!

Un tentativo di affrontare (stavo per dire: risolvere) il problema, può esser rappresentato dai tre programmi che compaiono in queste pagine e che descrivo uno per volta.

Il primo programma

Partiremo dal presupposto secondo cui il risultato di un incontro sportivo rappresenta (a parte la fortuna o la sfortuna che è impossibile predeterminare) da un gruppo di fattori concomitanti.

Alcune squadre, ad esempio, sono più valide in difesa anzichè in attacco; altre, invece, nei giochi fuori casa rendono molto meno; altre ancora giocano molto male in condizioni atmosferiche avverse mentre eventi negativi (totonero, figli presunti, ritiro passaporti, litigi con dirigenti) influiscono sul morale della squadra o di alcuni giocatori in modo più o meno determinante al fine di una valida partita.

Dato il Run al primo programma compare la domanda "Fattore campo?" cui risponderete con un numero, compreso tra 0 e 10, che dovrebbe essere il più obiettivo possibile. A tale scopo potete controllare (esaminando il comportamento della squadra in oggetto) i risultati ottenuti nel passato in quel campo di calcio ed assegnando un voto alto, medio o basso in caso di molte, poche vittorie o numerose sconfitte.

Con analoghi criteri risponderete alle domande successive, vale a dire assegnando un voto alla valutazione della difesa e dell'attacco, nel loro insieme; alla validità della presenza di tifosi (e della loro comprovata "efficienza", o meno) nella partita da disputare; al comportamento abituale dei giocatori nelle presumibili condizioni atmosferiche; alla valutazione (piuttosto aleatoria, lo ammetto) delle condizioni "medie" psicologiche e fisiche dell'intera squadra.

Il risultato dell'elaborazione non rappresenta altro che un voto medio, di scarsa utilità pratica.

Tale valutazione, infatti, deriva dall'attribuire l'identica importanza a fattori che, in realtà, influiscono in modo diverso sulle reali condizioni delle squadre. Un voto eventualmente basso assegnato alle condizioni atmosferiche (esempio: 3) viene controbilanciato a malapena da uno alto attribuito alle condizioni fisiche (esempio: 8) fornendo la modesta media del 5.5 [(8+3)/2].

Il programma, dunque, necessita di una "correzione". Con il secondo programma, pertanto, non solo apporteremo le modifiche necessarie, ma faremo in modo da rendere obiettivi, in seguito, i pronostici per le squadre di calcio.

Il secondo programma

Dato il Run, verrà chiesto di introdurre il numero di fattori determinanti. Supponiamo, per semplicità, di considerare solo efficienza fisica; effetto della tifoseria, condizioni psicologiche, effetto campo, valutazione difesa. Digiteremo, quindi, 5 come risposta.

Subito dopo verrà chiesto di digitare il numero delle squadre di calcio sulle quali siete in grado di esprimere un parere. Dovrete, almeno, inserire dati su tutte le squadre di serie A e B. Se non vi ritenete all'altezza nascondete tutti gli oggetti contundenti ed invitate a casa vostra tifosi di opposte fazioni allo scopo di giungere, vivi, alla determinazione di una valutazione obiettiva il più possibile.

Per comprendere più facilmente l'uso del programma proposto, rispondete 4 alla richiesta sul numero delle squadre.

In seguito digitate sia i fattori sia il nome delle squadre:

Fattore 1? Eff. fisica [R]
Fattore 2? Tifoseria [R]
Fattore 3? Cond.psicologiche [R]
Fattore 4? Effetto campo [R]
Fattore 5? Valut. difesa [R]

Alla richiesta di conferma premete S oppure N a seconda se siete soddisfatti dei dati digitati.

Analogamente rispondete alle domande successive, relative ai nomi delle squadre, digitando:

Squadra n.1? Alfa [R]
Squadra n.2? Beta [R]
Squadra n.3? Gamma [R]
Squadra n.4? Delta [R]

(Ho preferito evitare nomi di squadre vere, per evitare polemiche sui commenti che tra poco sarò costretto a fare...)

Dopo aver risposto affermativamente anche alla seconda domanda di conferma, verranno visualizzati uno per uno, per ciascuna squadra, i cinque fattori determinanti prima indicati. Sarà ora necessario assegnare, per ciascuno di essi, la percentuale di influenza sulla valutazione complessiva. Seguitemi nell'esempio per meglio comprendere:

Squadra: Alfa

Eff. fisica per%? 15[R] 85
Tifoseria per%? 43[R] 42
Cond.psicol. per%? 20[R] 22
Effet.campo per%? 10[R] 12

Il programma, in altre parole, assegna un totale di 100 alla globalità dei cinque fattori. Se ritenete che l'efficienza fisica valga al massimo 15 per la squadra Alfa, non appena premete il tasto Return verrà visualizzato il valore residuo 85 (100-15=85). Se, ancora, la tifoseria ha un ruolo importante per il risultato dell'incontro e decidete per un 43%, rimane a disposizione il 42% (opportunamente visualizzato) da distribuire tra i fattori rimanenti. Per l'ultimo fattore (Valut. difesa) la valutazione percentuale è

automatica, dal momento che la somma "deve" essere 100.

Anche in questo caso, confermando la digitazione, si passa alla squadra successiva (Beta) fino all'ultima (Gamma).

Al termine dell'assegnazione dovrete decidere se registrare i dati su cassetta o disco. Compiuta l'operazione non resta che utilizzare il terzo programma.

Come utilizzare il terzo listato

Dato il solito Run, rispondete in modo coerente alle varie domande (nastro o disco, nome del file); subito verranno caricati, e visualizzati, sia i nomi dei fattori che quelli delle squadre. Da notare che le percentuali vengono tenute nascoste.

Alla domanda "Squadra?" rispondete col nome desiderato, oppure con le prime lettere se non c'è pericolo di equivoco. digitate, dunque, alfa (oppure "al") e premete Return.

Comparirà il nome della squadra (per intero, in reverse) e la richiesta del voto da assegnare ai cinque fattori, che verranno visualizzati in successione. Supponiamo che la squadra Alfa giochi in casa, sul cui campo risulta imbattuta da tempo, e che la tifoseria è validissima: supporremo, inoltre, che la squadra sia da poco tempo uscita vittoriosa da una partita extra-campionato e che, purtroppo, non solo risente della fatica, ma deve rinunciare ad un valido difensore per proibizione del medico della società. Un'assegnazione dei voti può essere, in questo caso, la seguente:

Squadra Alfa
Eff. fisica voto? 5 [R]
Tifoseria voto? 10[R]
Cond.psicol.voto? 7 [R]
Effet.campo voto? 10[R]
Val. difesa voto? 6 [R]

L'elaborazione risulta 8.17 mentre la semplice media aritmetica porterebbe al valore 7.6. Ciò è accaduto proprio perchè ciascun voto è stato "pesato" con le percentuali impostate precedentemente. Il vantaggio di non poter ricordare tutte le percentuali impostate per tutte le squadre dovrebbe rendere l'elaborazione del voto finale più obiettiva.

Analogamente procederete alla determinazione del voto della squadra con cui Alfa dovrà battersi. A seconda delle elaborazioni, dunque, saprete assegnare con maggior equilibrio una vittoria, una sconfitta o un pareggio.

Non resta che invitare a casa vostra i consulenti per decidere le percentuali ma, mi raccomando, sceglieteli tra persone mature, responsabili e non litigiose...

```
100 REM DETERMINAZIONE DELLE
110 REM CONDIZIONI IN CUI
120 REM GIOCHERA' UNA SQUADRA
130 :
140 FOR I=0 TO 6:READ A$(I),A(I
    ):NEXT
150 :
160 PRINTCHR$(147)"ASSEGNA UN V
    OTO TRA 0 E 10":PRINT
170 FOR I=0 TO 6:PRINTA$(I);:IN
    PUT ":VOTO";VO
180 V=V+VO*A(I)/100:NEXT:PRINT:
    PRINT"VOTO TOTALE:"V
185 :
190 DATA "FATTORE CAMPO",20
200 DATA "CONDIZ.DIFESA",15
210 DATA "CONDI.ATTACCO",30
220 DATA "FATTORE  TIFO",7
230 DATA "CONDI.ATMOSF.",3
240 DATA "CONDI.PSICOL.",10
250 DATA "CONDI.FISICHE",15
```

```
100 REM PROGRAMMA DI IMPOSTAZIO
    NE
110 REM COEFFICIENTI
120 REM PER LA DETERMINAZIONE
130 REM DELL'EFFICIENZA DI
140 REM SQUADRE DI CALCIO
150 :
160 CL$=CHR$(147)
170 PRINTCL$"INTRODUCI UNO ALLA
     VOLTA I VARI FATTORI DETER
    MINANTI ";
180 PRINT"PER L'ESITO DI UN INC
    ONTRO DI CALCIO"
190 PRINT:INPUT "QUANTI FATTORI
    ";FA:IF FA<1 THEN 190
200 PRINT:INPUT "QUANTE SQUADRE
    ";NS:IF NS<1 THEN 200
210 DIM A$(FA):DIM B$(NS):DIM X
    (FA,NS)
220 FOR I=1 TO 20:A$=A$+"-":NEX
    T:PRINT
230 X$="":PRINTCL$:FOR I=1 TO F
    A
235 PRINT"FATTORE N."I;:INPUT A
    $(I):IF A$(I)="" THEN 235
240 A$(I)=LEFT$(A$(I)+A$,13):NE
    XT
250 PRINTCL$;:FOR I=1 TO FA:PRI
    NT"FATT.N."I;CHR$(18)A$(I):
    NEXT
260 PRINT:GOSUB 540:PRINT:PRINT
    :IF A$<>"S" THEN  RUN
270 PRINTCL$:FOR I=1 TO NS
280 B$(I)="":PRINT"SQUADRA N."I
    ;:INPUT B$(I):IF B$(I)="" T
```

```
    HEN 280
300 NEXT:PRINTCL$
310 FOR I=1 TO NS:PRINT"SQUADRA
     N."I;B$(I):NEXT
320 PRINT:GOSUB 540:IF A$<>"S"
    THEN 270
330 FOR X=1 TO NS
340 PRINTCL$"SQUADRA: "B$(X):PR
    INT
350 PX=100:FOR Y=1 TO FA-1
360 PRINTA$(Y);:PR=0:INPUT "PER
    .%";PR:PP=PX-PR
370 IF PP<0 THEN 360
380 PRINTCHR$(145) TAB(23)PP:PX
    =PP:X(Y,X)=PR:NEXT
390 X(FA,X)=PX
400 PRINTCL$"SQUADRA: "B$(X):PR
    INT:FOR Z=1 TO FA:PRINTA$(Z
    )X(Z,X):NEXT
410 PRINT:GOSUB 540:IF A$<>"S"
    THEN 340
420 NEXTX:PRINTCL$
430 PRINT"MEMORIZZAZ. DATI SU S
    UPPORTO MAGNETICO":PRINT
440 DN=1:PRINT"DISCO O NASTRO (
    D/N)":GOSUB 550
450 PRINT:PRINT:IF A$="D" THEN
    DN=8
460 PRINT"NOME DEL FILE DA ASSE
    GNARE AI FATTORI":INPUT A$:
    A$=A$+",S,W"
470 OPEN 1,DN,8,A$
480 PRINT#1,FA:REM SCRIVE N.FAT
    TORI
490 PRINT#1,NS:REM SCRIVE N.SQU
    ADRE
500 FOR X=1 TO FA:PRINT#1,A$(X)
    :NEXT
510 FOR X=1 TO NS:PRINT#1,B$(X)
    :NEXT
520 FOR X=1 TO FA:FOR Y=1 TO NS
    :PRINT#1,X(X,Y):NEXTY,X
530 CLOSE 1:PRINT"FINE LAVORO":
    END
540 PRINT"CONFERMI? (S/N)"
550 GET A$:IF A$="" THEN 550
560 RETURN
```

```
100 REM PROGRAMMA PER ELABORARE
     L'EFFICIENZA DI SQUADRE DI
     CALCIO
110 REM IN BASE A COEFFICIENTI
    PRECEDENTEMENTE IMPOSTATI
120 :
130 CL$=CHR$(147):RE$=CHR$(18)
140 PRINTCL$:DN=1:INPUT "NASTRO
     O DISCO (N/D)";A$:IF A$<>"
    N" THEN DN=8
150 A$="":PRINT"NOME DEL FILE D
    A CARICARE":INPUT A$:B$=A$:
    IF DN<>8 THEN 170
155 IF A$="" THEN  RUN
160 A$=A$+",S,R":PRINTCL$
170 OPEN 1,DN,0,A$
180 INPUT#1,FA:REM LEGGE N.FATT
    ORI
190 INPUT#1,NS:REM LEGGE N.SQUA
    DRE
200 DIM A$(FA):DIM B$(NS):DIM X
    (FA,NS)
210 PRINT"N.FATTORI DEL FILE "B
    $"="FA:PRINT
220 FOR X=1 TO FA:INPUT#1,A$(X)
    :PRINTRE$A$(X),:NEXT:PRINT
230 PRINT"N.SQUADRE DEL FILE "B
    $"="NS:PRINT
240 FOR X=1 TO NS:INPUT#1,B$(X)
    :PRINTRE$B$(X),:NEXT
250 FOR X=1 TO FA:FOR Y=1 TO NS
    :INPUT#1,X(X,Y):NEXTY,X
260 CLOSE 1:PRINT:PRINT:GOSUB 3
    40
270 PRINTCL$:SQ$="":INPUT "SQUA
    DRA";SQ$:LE=LEN(SQ$):IF SQ$
    ="" THEN 270
280 DV=0:FOR I=1 TO NS:IF LEFT$
    (B$(I),LE)=SQ$ THEN DV=1:SI
    =I:I=NS
290 NEXT:IF DV=0 THEN PRINT:PRI
    NT"SQUADRA INESISTENTE":GOS
    UB 340:GOTO 270
300 PRINTCL$RE$B$(SI):PRINT:TT=
    0
310 FOR I=1 TO FA:PRINTA$(I);:V
    T=0:INPUT "VOTO";VT
320 TT=TT+X(I,SI)*VT/100:NEXT
330 PRINT:PRINT"VOTO ELABORATO:
    "RE$TT:GOSUB 340:GOTO 270
340 PRINT:PRINT"PER CONTINUARE
    PREMI UN TASTO"
350 GET A$:IF A$="" THEN 350
360 RETURN
```

Gioco innocente...

Un programma brevissimo e semplice che simula un passatempo di antica memoria

di Maurizio Dell'Abate

Pochissime spiegazioni per questo programma che, per la sua brevità, merita di esser copiato ed utilizzato per stupire gli amici che ancora non possiedono un computer.

Le spiegazioni indispensabili per giocare verranno visualizzate mediante istruzioni Print.

Quando siete stufi e volete scoprire il trucco, esaminate in dettaglio il listato: in pochi minuti (giorni?) riuscirete ad eseguire lo stesso gioco con 21 carte, e i vostri amici (quelli di prima) continueranno a stupirsi (cambiate amici: forse sono un po' tonti).

"E' un gioco vecchio!" diranno i maligni che, però (cfr. "La volpe e l'uva"), non sono stati in grado di "trasferirlo" su computer...

```
100 REM  GIOCO MAGICO - BY M. D
    ELL'ABATE
110 REM  PER QUALSIASI COMPUTER
120 :
130 PRINTCHR$(147)
140 PRINT TAB(12);"COMPUTER MAG
    ICO"
150 DIM A$(20)
160 A$="MREILNFPUGTQAHZBOCVDS"
170 IF RND(1)<.33 THEN A$="LSCM
    TARDVNBOPQFUEZGIH"
180 IF RND(1)<.66 THEN A$="ALBM
    CNDOEPFQGRHSITUZV"
190 XB=LEN(A$)+900+17*2
200 FOR G=0 TO 1000:NEXT:PRINTC
    HR$(147)
210 PRINT" HAI 10 SECONDI PER P
    ENSARE UNA"
220 PRINT" LETTERA DALLA A ALLA
     Z DELL'ALFABETO"
230 PRINT" ITALIANO (ABCDEFGHIL
    MNOPQRSTUVZ)."
240 PRINT" IO NON LA DEVO SAPER
    E!!"
250 FOR I=0 TO 5000:NEXT
260 FOR I=1 TO 21:A$(I-1)=MID$(
    A$,I,1):NEXT
270 IF CT=3 THEN 450
280 GOSUB 310
290 PRINTCHR$(147):FOR CA=0 TO
    500:NEXT
300 GOTO 260
310 PRINTTCHR$(147):PRINTSPC(1)
    ;"-COL 1--COL 2--COL 3-"
320 PRINT:PRINT:PRINT:PRINT
330 FOR I=0 TO 20 STEP 3
340 PRINTSPC(4);A$(I);SPC(6);A$
    (I+1);SPC(6);A$(I+2)
350 PRINT:NEXT
360 PRINT:PRINT" IN CHE COLONNA
     SI TROVA? (1-2-3)"
370 GET Z$:IF Z$<"1" OR Z$>"3"
    THEN 370
380 B$="":FOR Z=0 TO 18 STEP 3:
    B$=B$+A$(Z):NEXT
390 N$="":FOR Z=1 TO 19 STEP 3:
    N$=N$+A$(Z):NEXT
400 M$="":FOR Z=2 TO 20 STEP 3:
    M$=M$+A$(Z):NEXT
410 IF Z$="1" THEN A$=N$+B$+M$:
    GOTO 440
420 IF Z$="2" THEN A$=M$+N$+B$:
    GOTO 440
430 A$=N$+M$+B$
440 CT=CT+1:RETURN
450 PRINTCHR$(147)
460 ZX$=A$(SQR(1000-XB+40+4↑2-1
    ))
470 PRINT" HAI SCELTO:        ";Z
    X$
```

Matematica a gogo

a cura di Alessandro de Simone

15200 Divisioni con N decimali

Uno dei limiti delle calcolatrici tascabili, più o meno potenti che siano, è senza dubbio quello della precisione multipla. Per alcuni calcoli spesso non bastano le 10 o 13 cifre significative, ed a volte ne occorrono ben di più.

Il problema non è di difficile soluzione quando si tratta di addizioni o di sottrazioni, perchè in tal caso basta scomporre il numero in più parti (di 7 o 8 cifre ciascuna) ed eseguire l'operazione blocco per blocco, tenendo conto degli eventuali riporti; in seguito si ricomporrà la cifra, pronta per essere utilizzata.

Analogamente, per la moltiplicazione, spesso basta effettuare operazioni parziali, cifra per cifra, con l'ausilio della calcolatrice, e poi effettuare le somme, naturalmente tenendo conto della significatività della cifra; un po', insomma, come per una moltiplicazione fatta "a mano".

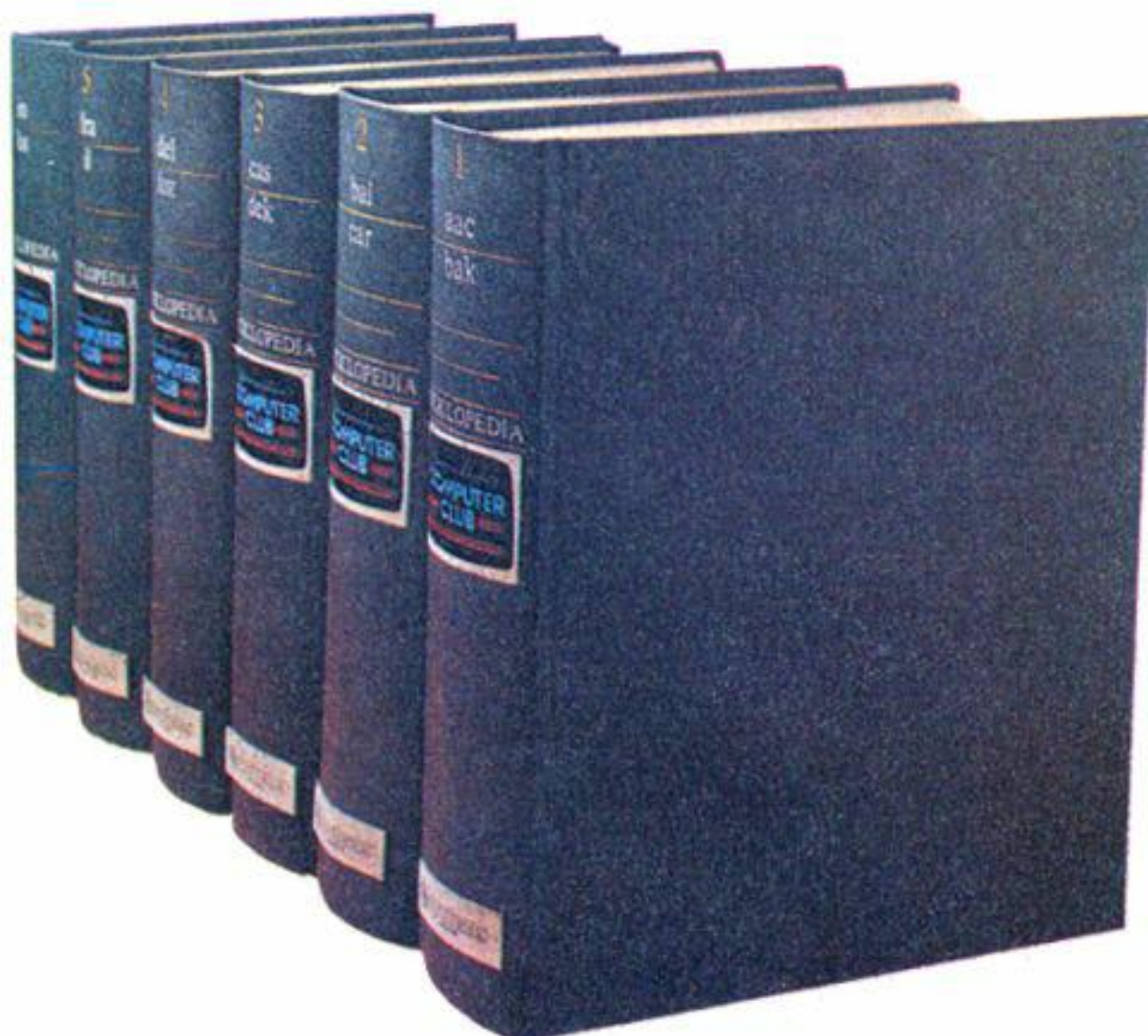
Ma il discorso si complica quando si parla di divisioni in multiprecisione, perchè per ogni decimale in più la mole di calcoli da svolgere non è indifferente. Ed ecco il nostro calcolatore venirci incontro per risolvere il problema: niente di più semplice, per lui!

Scherzi a parte, il programma presentato permette di effettuare divisioni con un numero di decimali limitato dalla sola capacità di macchina nel trattare le stringhe: tenendo conto della presenza di eventuali segni, della parte intera e del punto decimale, di solito è bene non superare i 250 decimali. Si tratta, ad ogni buon conto, di una precisione di tutto rispetto.

La "chiamata" alla subroutine 15200 avviene molto semplicemente passando come parametri il dividendo (in X0), il divisore (in Y0) ed il numero di decimali desiderato (in W0). Il risultato sarà espresso in forma di stringa (X1$), con le limitazioni di cui parlavamo prima.

Naturalmente eventuali errori saranno verificati tempestivamente e causeranno un'uscita anticipata dal sottoprogramma con la stringa X0$ impostata ad ERR.

L'algoritmo utilizzato è molto spartano, ma provvede egregiamente allo scopo: si tratta pari pari del procedimento che si usa per risolvere manualmente le divisioni. Ma provate un po' a vedere chi trova per primo il risultato!

```
100 PRINTCHR$(147)"DIMOSTRATIVO
    "
110 PRINT"DIVISIONI CON 'N' DEC
    IMALI"
120 PRINT:REM BY SIMONE BETTOLA
130 PRINT"IL QUOZIENTE DI 10/(-
    3.1)"
140 PRINT"CON 200 DECIMALI VALE
    ...";
150 :
200 X0=10:Y0=-3.1:W0=200
210 GOSUB 15200
220 PRINTX1$:PRINT:PRINTX0$
230 :
9999 END
15200 IF Y0=0 OR W0>250 OR W0<1 T
      HEN 15288
15205 IF INT(X0)<>X0 OR INT(Y0)<>
      Y0 THEN X0=X0*10:Y0=Y0*10:G
      OTO 15210
15210 W9= SGN(X0*Y0):X0=ABS(X0):Y
      0=ABS(Y0):W8=INT(X0/Y0):X1$
      =STR$(W8*W9)+"."
15215 IF W0>255-LEN(X1$) OR W0<1
      THEN 15288
15220 FOR W5=1 TO W0:X0=(X0-Y0*AB
      S(W8))*10:W8=INT(X0/Y0)
15225 X1$=X1$+RIGHT$(STR$(W8),LEN
      (STR$(W8))-1):NEXT:GOTO 152
      89
15288 X0$="ERR":RETURN
15289 X0$="OK":RETURN
15290 REM X0=DIVIDENDO (NUMERATOR
      E)
15291 REM Y0=DIVISORE (DENOMINATO
      RE)
15292 REM W0=NUMERO DI DECIMALI R
      ICHIESTO
15293 REM X1$=RISULTATO DIVISIONE
15299 REM NOME:DIVISIONE MULTIPRE
      CISIONE
```

15300 Semplificazione di frazioni

Siamo tornati sui banchi di scuola, e la maestra ci chiama alla lavagna per semplificare una frazione... No, niente di tutto questo, ma il programma può dare una mano in casi disperati.

Il problema ricorrente, quando si utilizzano calcolatrici per i compiti di scuola, è che il risultato viene espresso direttamente in notazione decimale con una sfilza di cifre dopo la virgola. Il calcolatore può venire in aiuto, ma sfortunatamente, con la precisione a nostra disposizione, il Commodore 64 non riuscirebbe nel nostro intento.

A volte basta semplificare astruse frazioni per rendersi conto che un determinato calcolo è ben più semplice di quanto non si credesse a prima vista. Ed in questo il programmino di queste pagine è veramente formidabile. In men che non si dica siamo in grado di semplificare una frazione, di conoscere i valori per cui è stata semplificata e perfino i passaggi intermedi.

La tecnica utilizzata per ricavare i termini della semplificazione è certamente brutale, ma funziona allo scopo: non era certo il caso di fare i raffinati a discapito della velocità.

In X1 va posto il numeratore ed in Y1 il denominatore della frazione da semplificare, variabili in cui sarà anche riportato il risultato finale. La variabile X0 è invece un flag, nel senso che, se è impostato ad uno, la subroutine effettua la stampa dei termini intermedi, mentre se è uguale a zero, il programma terminerà direttamente con il risultato.

```
 100 PRINTCHR$(147)"DIMOSTRATIVO
     "
 110 PRINT"SEMPLIFICAZIONE DI FR
     AZIONI"
 120 PRINT:REM BY SIMONE BETTOLA
 130 PRINT"SEMPLIFICA LA FRAZION
     E[RVS] 15/600 "
 140 PRINT"E NE VISUALIZZA I TER
     MINI INTERMEDI"
 150 :
 200 X0=1:X1=15:Y1=600
 210 GOSUB 15300
 220 PRINTX1;"/";Y1
 230 :
9999 END
15300 IF Y1=0 OR X0<0 OR X0>1 THE
      N 15388
15305 W0=2:W1=1:IF X1/Y1=INT(X1/Y
      1) THEN X1=X1/Y1:Y1=1:GOTO
      15389
15310 W2=X1/W0-INT(X1/W0):W3=Y1/W
      0-INT(Y1/W0):IF W2=0 AND W3
      =0 THEN 15320
15315 W0=W0+W1:W1=2:GOTO 15325
15320 X1=X1/W0:Y1=Y1/W0:IF X0=1 T
      HEN PRINTW0,X1;"/";Y1
15325 IF W0>X1 OR W0>Y1 THEN 1538
      9
15330 GOTO 15310
15388 X0$="ERR":RETURN
15389 X0$="OK":RETURN
15390 REM X1=NUMERATORE
15391 REM Y1=DENOMINATORE
15392 REM X0=FLAG(1=STAMPA TERMIN
      I INTERMEDI: 0=NESSUNA STAM
      PA)
15399 REM NOME: SEMPLIFICAZIONI
```

15400 Conversioni tra basi qualsiasi

Quante volte vi è capitato di convertire un numero da base 11 a base 7? Probabilmente mai!

E' bene chiarire che cosa significa il termine "In base N": si tratta di un raggruppamento di simboli numerici che hanno solamente 7 valori (compreso lo zero) invece dei dieci a cui siamo abituati. In base 7 dopo il 6 viene immediatamente 10 (che si legge "uno zero" e non "dieci"), che però, in base 10, vale sempre 7.

Anche se in apparenza il discorso sembra un po' contorto e senza dubbio complicato, tentiamo di capire come funziona la faccenda.

Supponiamo di appartenere ad una strana tribù di spaziali che ha solo 7 dita (!). Provando ad elencare sulle vostre dita i numeri possibili, avrete 0, 1, 2, 3, 4, 5, 6 e poi...? Dovreste riprendere da 0 a patto di considerare un valore di riporto, che si chiamerà "settina" (invece di decina) e poi ancore la "settantina" (per gli umani centinaia) e via di seguito.

Naturalmente il discorso è campato in aria, anche perchè sarebbe difficile pensare ad esseri con una mano da quattro dita e l'altra da tre!

Comunque è bene osservare che non a caso l'Homo Sapiens conta in base 10 ed ha per l'appunto 10 dita (o è il contrario?...).

Ma allontaniamoci da disquisizioni di ordine antropologico per analizzare più concretamente il problema dal punto di vista matematico.

Sul nostro computer una POKE può avere come argomento un valore massimo di 255 e il motivo è da reperirsi nel fatto che il microprocessore tratta 8 bit. Che cosa c'entra questo fatto con il 255 ? Utilizzando il programma in questione, si converte da base binaria (che è quella del calcolatore) una serie di otto "1" (%=11111111) in base dieci (che poi è la nostra). Per i curiosoni si può anche tentare di convertire in base 16, ma a voi la sorpresa.

Relativamente al programma, sono da sottolineare alcuni punti. Innanzitutto, per basi superiori a dieci, si ricorre alle lettere dell'alfabeto inglese, e quindi dopo la

"I" vi sarà la "J". Ovviamente la massima base possibile sarà 37, corrispondente alla "Z". Appositi controlli consentono di dimensionarli a seconda della base utilizzata, mentre per motivi facilmente intuibili i valori iniziali e finali saranno contenuti in stringhe.

Per evitare problemi strani, è possibile utilizzare solo numeri positivi, dal momento che la notazione in complemento a due crea non pochi problemi e non è di facile comprensione.

```
  100 PRINTCHR$(147)"DIMOSTRATIVO
      "
  110 PRINT"CONVERSIONI TRA BASI
      QUALSIASI"
  120 PRINT:REM BY SIMONE BETTOLA
  130 PRINT"[RVS]190A7[RVOFF] IN
      BASE 11"
  140 PRINT"E' UGUALE, IN BASE 7,
       A"
  150 :
  200 X1=11:Y1=7:X1$="190A7"
  210 GOSUB 15400
  220 PRINTY1$
  230 :
 9999 END
15400 W0=0:IF X1<2 OR Y1<2 OR X1>
      37 OR Y1>37 THEN 15488
15405 IF LEN(X1$)>INT(LOG(1.7E38)
      /LOG(X1))-1 THEN 15488
15410 FOR W9=LEN(X1$) TO 1 STEP -
      1:W1=ASC(MID$(X1$,W9,1)):IF
       W1>64 THEN W1=W1-7
15415 W1=W1-48:IF W1>=X1 THEN 154
      88
15420 W0=W0+W1*X1↑(LEN(X1$)-W9):N
      EXT
15425 Y1$="":W8=INT(LOG(W0)/LOG(Y
      1)):FOR W9=0 TO W8:W1=W0-IN
      T(W0/Y1)*Y1
15430 W0=INT(W0)/Y1:IF W1>9 THEN
      W1=W1+7
15435 W1=W1+48:Y1$=CHR$(W1)+Y1$:N
      EXT:GOTO 15489
15488 X0$="ERR":RETURN
15489 X0$="OK":RETURN
15490 REM X1=BASE INIZIALE
15491 REM Y1=BASE VOLUTA
15492 REM X1$=VALORE DA CONVERTIR
      E
15493 REM Y1$=VALORE NELLA NUOVA
      BASE
15499 REM NOME: CONVERSIONE TRA B
      ASI
```

15500 Logaritmi in base qualsiasi

Questa brevissima routine sarà certamente utile a tutti coloro che desiderano realizzare un buon programma di risoluzione di equazioni o di studio di funzioni.

Calcola il logaritmo di un numero in una base qualsiasi, a patto che numero e base rispettino il campo di esistenza dei logaritmi stessi, altrimenti si otterrà un X0$="ERR".

L'argomento del logaritmo, memorizzato in X0, dovrà essere positivo e non nullo; la base del logaritmo sarà memorizzata in Y0 e dovrà essere un numero positivo, non nullo e diverso da uno. Entrambi dovranno essere numeri reali non necessariamente interi.

Il risultato del logaritmo, il valore cioè a cui bisognerebbe elevare la base per ottenere come risultato l'argomento del logaritmo, verrà restituito in X1 e potrà assumere un qualsiasi valore reale anche negativo o nullo.

Il programma fa uso di una delle principali proprietà dei logaritmi. Chi è particolarmente interessato, potrà approfondire l'argomento su un qualsiasi testo di algebra per le scuole superiori.

Per ciò che riguarda un utilizzo più computeristico della routine, è interessante notare che il logaritmo di un numero intero in una base qualsiasi addizionato ad uno, corrisponde esattamente al numero di cifre che occorrono per rappresentare quel numero nel sistema numerico che ha per base la base del logaritmo (Help!).

Ricorriamo ad un esempio (in base 10):
$\log 10000 = 4$
10000 è appunto formato da 5 (4+1) cifre. Questa proprietà risulta molto utile quando occorre convertire un numero da una base numerica ad un'altra, cosa molto frequente nel campo dei computer.

```
  100 PRINTCHR$(147)"DIMOSTRATIVO
      "
  110 PRINT"LOGARITMI IN BASE QUA
      LSIASI"
  120 PRINT:REM BY SILVIO CALLEGA
      RI
  130 PRINT"IL LOGARITMO DI 65536
       IN BASE 2 E'"
  140 :
  200 X0=65536:Y0=2
  210 GOSUB 15500
  220 PRINTX1
  230 :
 9999 END
15500 IF Y0<=0 OR Y0=1 OR X0<=0 T
      HEN 15588
15510 X1=LOG(X0)/LOG(Y0)
15520 GOTO 15589
15588 X0$="ERR":RETURN
```

```
15589 X0$="OK":RETURN
15590 REM X0: ARGOMENTO DEL LOGAR
      ITMO
15592 REM Y0: BASE DEL LOGARITMO
15594 REM X1: RISULTATO DEL LOGAR
      ITMO
15599 REM NOME: LOGARITMI IN BASE
      N
```

15600 Conversione di coordinate

Oggi si usa molto spesso la rappresentazione di dati sotto forma grafica come si può notare, tra l'altro, leggendo le pagine finanziarie dei quotidiani. Questo perchè i grafici riescono a fornire molto più facilmente l'idea delle differenze e dei nessi fra i dati che rappresentano.

La prima rappresentazione grafica realizzata è quella che fa riferimento ad un piano al quale appartengono i punti che rappresentano i dati.

Vi sono due modi per indicare un punto del piano. Il primo è il sistema di coordinate cartesiane (dal nome del loro inventore Descartes) nel quale il piano è caratterizzato da due assi ortogonali (cartesiani anch'essi) che si incrociano in un punto chiamato origine degli assi.

Le coordinate del punto sono date dalle proiezioni del punto stesso sugli assi.

Il secondo sistema è chiamato di coordinate polari ed è costituito da un'origine e da una semiretta che, partendo dall'origine, si dirige verso la destra di un osservatore che guardi perpendicolarmente il piano. In questo un punto è identificato dalla lunghezza del segmento che congiunge il punto con l'origine e dall'angolo compreso tra quest'ultimo e la semiretta di riferimento.

E' da notare che, in tale rappresentazione, un punto può essere indicato da più valori dell'angolo, tutti sfasati di un giro (2*pigreco) rispetto al precedente.

Di solito, però, si utilizzano i valori compresi tra 0 a pigreco (per angoli misurati in senso antiorario) e i valori da 0 a -pigreco (per gli angoli misurati in senso orario).

La routine proposta è in grado di effettuare la conversione da coordinate cartesiane a polari, e viceversa.

Le variabili da definire prima di "chiamare" la routine sono:

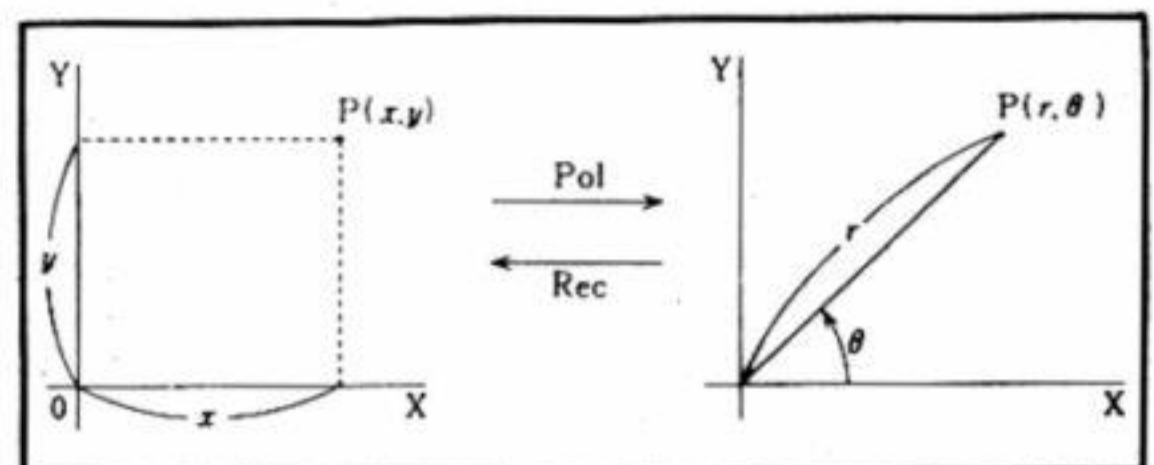
X9=0 per convertire da cartesiane a polari.

P9=1 per convertire da polari a cartesiane.

X0 contiene il valore dell'ascissa se la conversione è da cartesiane a polari, altrimenti rappresenta la lunghezza del segmento.

Y0 contiene il valore dell'ordinata, nel primo caso e l'ampiezza dell'angolo nel secondo.

Le variabili possono assumere qualunque valore reale tenendo conto che l'unica limitazione è imposta dal range numerico del computer.

```
  100 PRINTCHR$(147)"DIMOSTRATIVO
      "
  110 PRINT"CONVERSIONE DA COORDI
      NATE CARTESIANE A"
  120 PRINT"POLARI E VICEVERSA"
  130 PRINT:REM BY SILVIO CALLEGA
      RI
  140 PRINT"R=23","O=[PI]/2"
  150 PRINT
  160 :
  200 X0=23:Y0=π/2:X9=1
  210 GOSUB 15600
  220 PRINT"X="X1,"Y="Y1
  230 :
 9999 END
15600 IF X9<>0 AND X9<>1 THEN 156
      88
15610 IF X9=1 THEN 15650
15620 IF X0=0 THEN X1=ABS(Y0):Y1=
      SGN(Y0)*π/2:GOTO 15689
15630 IF Y0=0 AND X0<0 THEN X1=AB
      S(X0):Y1=π:GOTO 15689
15640 X1=SQR(X0*X0+Y0*Y0):Y1=ATN(
      Y0/X0):GOTO 15689
15650 X1=X0*COS(Y0):Y1=X0*SIN(Y0)
      :GOTO 15689
15688 X0$="ERR":RETURN
15689 X0$="OK":RETURN
15690 REM X0,Y0: VALORI DELLE COO
      RDINATE INIZIALI
15692 REM X9=1: CONVERSIONE POLAR
      I->CARTESIANE
15694 REM X9=0: CONVERSIONE CARTE
      SIANE->POLARI
15696 REM X1,Y1: COORDINATE RISUL
      TANTI
15699 REM NOME: CONVERSIONE COORD
      INATE
```

Ed è subito Sprite!

Qualche routine per la gestione degli sprite che verrà sicuramente apprezzata dagli appassionati del Simon's Basic

di S.Bettola e S.Callegari

Definizione sprite (21261/21473)

Finalmente è disponibile anche nel Basic 2.0 una funzione per definire gli sprite che molti avranno già avuto modo di utilizzare sul famoso Simon's Basic: niente più complicati calcoli per convertire un disegno in dati comprensibili al computer, ma solo righe Basic di semplice gestione.

La routine di queste pagine permette infatti di definire lo sprite inserendo direttamente all'interno di una stringa i punti che formeranno la figura.

Ad ogni punto colorato dovrà corrispondere un asterisco (il segno comunemente usato per la moltiplicazione) mentre qualsiasi altro carattere, spazio compreso, corrisponderà ad uno spazio vuoto.

Abbiamo parlato di una stringa, ma in realtà si tratta di un gruppo di 21 stringhe tutte, però, con lo stesso nome ed una lunghezza di 24 caratteri ciascuna: uno sprite misura appunto 24x21 pixel.

Per utilizzare correttamente la routine bisogna tenere presente che:

- **il nome della stringa deve essere formato da due lettere e non da una lettera e un numero (andranno benissimo AB$, CD$, EF$, ma non A1$ oppure D7$).**
- **ogni nome di stringa dovrà essere utilizzato esclusivamente per definire il proprio sprite e non potrà essere riutilizzato in altre parti del programma o per altri sprite.**
- **le 21 stringhe contenenti il disegno dovranno trovarsi su linee di programma consecutive, non dovrà cioè essere presente alcuna istruzione tra due o più di esse.**
- **ciascuna stringa dovrà essere lunga 24 caratteri ed essere la sola istruzione della linea Basic di programma; non dovrà essere preceduta da altre istruzioni nè dall'assegnazione LET.**

Se una sola di queste norme non verrà rispettata la routine potrebbe segnalare un errore (Syntax oppure Illegal Quantity).

Oltre al nome della stringa contenente i dati, occorrerà "passare" alla routine l'indirizzo della locazione di partenza del blocco in cui memorizzare i dati.

N.B. Non importa se vengono prima definite le stringhe o chiamata la routine. Quest'ultima provvederà infatti a ricercare all'interno del programma la prima riga dei dati utili.

Potete quindi sistemare i disegni in coda al programma come fareste solitamente con normali istruzioni DATA.

```
1000 PRINTCHR$(147)"QUESTA ROUTI
     NE CONSENTE DI DEFINIRE UNO
      SPRITE ";
1010 PRINT"DISEGNANDOLO DIRETTAM
     ENTE ALLO   INTERNO DI UNA
     STRINGA DEL BASIC.";
1020 PRINT" ALLA ROUTINE DEVONO
     ESSERE COMUNICATI IL NOME D
     ELLA ";
1030 PRINT"STRINGA E L'INDIRIZZO
      DOVE MEMORIZZARE I DATI."
1040 RETURN
1050 DATA 032,253,174,032,019,17
     7,144,115,133,251
1060 DATA 032,115,000,032,019,17
     7,144,105,133,252
1070 DATA 032,115,000,201,036,20
     8,093,032,115,000
1080 DATA 032,253,174,032,138,17
     3,032,247,183,169
1090 DATA 021,133,099,165,043,13
     3,253,165,044,133
1100 DATA 254,160,000,177,253,13
     3,097,200,177,253
1110 DATA 240,050,133,098,200,20
     0,200,177,253,197
1120 DATA 251,208,028,200,177,25
     3,197,252,208,021
1130 DATA 200,177,253,201,036,20
     8,014,200,177,253
```

```
1140 DATA 201,178,208,029,200,17
     7,253,201,034,240
1150 DATA 043,165,097,133,253,16
     5,098,133,254,024
1160 DATA 144,195,165,099,208,00
     1,096,162,014,044
1170 DATA 162,022,044,162,011,07
     6,055,164,162,014
1180 DATA 160,002,177,253,133,05
     7,200,177,253,133
1190 DATA 058,076,055,164,165,09
     9,240,236,162,024
1200 DATA 200,177,253,240,229,20
     1,034,240,225,201
1210 DATA 042,240,003,024,144,00
     1,056,038,102,038
1220 DATA 101,038,100,202,208,23
     0,200,17-,253,240
1230 DATA 203,201,034,208,199,16
     0,002,185,100,000
1240 DATA 145,020,136,016,248,02
     4,165,020,105,003
1250 DATA 133,020,169,000,101,02
     1,133,021,198,099
1260 DATA 024,144,144,-1,28487
```

```
100 REM DEMO DEFINIZIONE SPRITE
110 X=21261
120 REM INDIRIZZO SUGGERITO SU CCC
130 AB$=".......**********......."
140 AB$="......************......"
150 AB$=".....**************....."
160 AB$="....*****AAAAAA*****...."
170 AB$="...******************..."
180 AB$="...******************..."
190 AB$="...*******....*******..."
200 AB$="...******************..."
210 AB$="...******************..."
220 AB$="....****************...."
230 AB$=".....**************....."
240 AB$="......************......"
250 AB$="......**..****..**......"
260 AB$=".....**...****...**....."
270 AB$="....**..********..**...."
280 AB$="...**...********...**..."
290 AB$="..**................**.."
300 AB$="******............******"
310 AB$="........................"
320 AB$="........................"
330 AB$="........................"
340 SYSX,AB$,832
350 POKE 2040,13:POKE 53269,1
360 POKE 53248,100:POKE 53249,100
370 POKE 53287,1
380 PRINTCHR$(147)"QUESTO E' UNO"
390 PRINT"SPRITE DEFINITO TRAMITE"
400 PRINT"UNA STRINGA BASIC"
```

Sprite tool (21474/21839)

Come è certamente noto a tutti, il Commodore 64 ha possibilità grafiche di indubbio pregio, tra cui la gestione contemporanea di 8 sprite. Ma è anche vero che per gestire le animazioni dobbiamo fare ricorso ad un numero considerevole di istruzioni POKE, con argomenti che vanno calcolati di volta in volta e.... beh, non tutti riusciamo a pensare in binario!

Ed ecco il linguaggio macchina venirci incontro amichevolmente, consentendo la realizzazione di un tool kit completo per la gestione degli sprite.

Ma, purtroppo, anche il Linguaggio Macchina ha i suoi limiti, che, nel nostro caso, consistono in un gruppo di ben 366 codici utilizzati per creare l'utility di queste pagine. Una volta digitata, però, la routine diventerà indispensabile per animare i vostri giochi, rendendoli sempre più avvincenti e divertenti.

Per ciò che riguarda il programma, c'è da dire che per renderlo rilocabile non è stato possibile utilizzare subroutine, costringendoci a ripetere parecchie volte gli stessi gruppi di istruzioni.

Chi ha una conoscenza approfondita del L.M. non incontrerà certo difficoltà a ridurre la mole del programma, sacrificandone, però, la rilocabilità.

Il folto gruppo degli argomenti della SYS andrà separato sempre e solo da una virgola e dovrà sempre essere riportato integralmente: non è possibile tralasciare alcun dato, anche se non interessa. Gli argomenti, da indicare nell'ordine, sono:

- **Numero dello sprite (da 0 a 7)**
- **Sua attivazione [on/off] (1/0)**

Se si sceglie di disattivare uno sprite, ma solo in questo caso, si possono, anzi, si devono tralasciare i rimanenti parametri:

- **Coordinata X (da 0 a 512)**
- Coordinata Y (da 0 a 255)
- Espansione X [on/off] (1/0)
- Espansione Y [on/off] (1/0)
- Multicolor [on/off] (1/0)
- Trasparenza [on/off] (1/0)
- Blocco (da 0 a 255)

Il blocco di memoria da cui traggono informazioni gli sprite è relativo al blocco principale del circuito VIC,

che è in grado di esplorare solo 16 Kbyte di memoria per volta. La locazione a partire dalla quale verranno memorizzati i dati sarà data da:

Locaz. = Vic*16384 + Blocco*64

Normalmente il VIC è posizionato sul blocco zero, dunque la formula si riduce a:

Locaz. = Blocco*64

Per approfondire in dettaglio la gestione delle animazioni, rimandiamo al N.35 di Commodore Computer Club ("Tutto sugli Sprite").

Come si può ampiamente dedurre dal programma dimostrativo, gli argomenti possono essere costituiti anche da variabili, che è conveniente chiamare con nomi che possano ricordare il significato degli argomenti, da specificare solo una volta e poi da modificare a proprio piacimento, secondo le velleità della vostra sbrigliata fantasia !

```
1000 PRINTCHR$(147)"SPRITE TOOL"
1001 PRINT"[2 DOWN]SYS XXXXX, PA
     RAMETRI (SEPARARE CON VIRGO
     LE. TUTTI SONO NECESSARI)"
1002 PRINT"[DOWN]NUMERO (0-7)"
1003 PRINT"SPRITE ON/OFF (1/0)"
1004 PRINT"COLORE (0-15)"
1005 PRINT"COORDINATA X (0-512)"
1006 PRINT"COORDINATA Y (0-255)"
1007 PRINT"MULTICOLOR ON/OFF (1/
     0)"
1008 PRINT"TRASPARENZA ON/OFF (1
     /0)"
1009 PRINT"BLOCCO (0-255)"
1099 RETURN
1100 DATA 032,253,174,032,138,17
     3,032,247,183,165
1110 DATA 020,201,008,016,043,13
     3,251,165,021,208
1120 DATA 037,169,001,166,251,24
     0,004,010,202,208
1130 DATA 252,133,253,073,255,13
     3,254,032,253,174
1140 DATA 032,138,173,032,247,18
     3,165,021,208,008
1150 DATA 165,020,240,007,201,00
     1,240,014,076,072
1160 DATA 178,165,254,045,021,20
     8,141,021,208,076
1170 DATA 248,168,032,253,174,03
     2,138,173,032,247
1180 DATA 183,165,020,201,016,01
     6,227,166,021,208
1190 DATA 223,166,251,157,039,20
     8,032,253,174,032
1200 DATA 138,173,032,247,183,16
     5,251,010,170,134
1210 DATA 252,165,020,157,000,20
     8,165,021,240,006
1220 DATA 201,001,240,013,208,18
     8,165,254,045,016
1230 DATA 208,141,016,208,024,14
     4,008,165,253,013
1240 DATA 016,208,141,016,208,03
     2,253,174,032,138
1250 DATA 173,032,247,183,166,02
     1,208,156,165,020
1260 DATA 166,252,157,001,208,03
     2,253,174,032,138
1270 DATA 173,032,247,183,165,02
     1,208,008,165,020
1280 DATA 240,007,201,001,240,01
     4,076,072,178,165
1290 DATA 254,045,029,208,141,02
     9,208,024,144,008
1300 DATA 165,253,013,029,208,14
     1,029,208,032,253
1310 DATA 174,032,138,173,032,24
     7,183,165,021,208
1320 DATA 221,165,020,240,006,20
     1,001,240,013,208
1330 DATA 211,165,254,045,023,20
     8,141,023,208,024
1340 DATA 144,008,165,253,013,02
     3,208,141,023,208
1350 DATA 032,253,174,032,138,17
     3,032,247,183,165
1360 DATA 021,208,179,165,020,24
     0,006,201,001,240
1370 DATA 013,208,169,165,254,04
     5,028,208,141,028
1380 DATA 208,024,144,008,165,25
     3,013,028,208,141
1390 DATA 028,208,032,253,174,03
     2,138,173,032,247
1400 DATA 183,165,021,208,137,16
     5,020,240,017,201
1410 DATA 001,240,002,208,048,16
     5,253,013,027,208
1420 DATA 141,027,208,024,144,00
     8,165,254,045,027
1430 DATA 208,141,027,208,032,25
     3,174,032,138,173
1440 DATA 032,247,183,166,021,20
     8,016,166,251,165
1450 DATA 020,157,248,007,165,25
     3,013,021,208,141
1460 DATA 021,208,096,076,072,17
     8,-1,47441
```

```
10 REM DIMOSTRATIVO DI SPRITE
   TOOL
100 X=21474:REM INDIRIZZO SUGGE
    RITO SU COMMODORE COMPUTER
    CLUB
110 PRINTCHR$(147):FOR I=1 TO 1
    0:PRINT"[DOWN]";:NEXT:PRINT
    "COMMODORE 64"
120 CY=130:BL=13:NU=0:MU=0:TS=0
    :EX=0:EY=0
130 FOR I=0 TO 63:POKE BL*64+I,
    255:NEXT
140 CO=INT(RND(0)*16)
150 FOR CX=0 TO 344 STEP 2
160 IF CX=160 THEN EX=1-EX:EY=1
    -EY:TS=1-TS
170 SYSX,NU,1,CO,CX,CY,EX,EY,MU
    ,TS,BL
180 NEXT:GOTO 140
```

Collisione (21840/21919)

La dinamicità dei giochi per C/64 che utilizzano gli sprite consiste nel fatto che è possibile rilevare eventuali contatti tra le stesse animazioni. Ma il compito non è sempre così semplice dal momento che bisogna, anche in questo caso, ricorrere all'istruzione PEEK. Se poi si desidera conoscere lo sprite con il quale è avvenuta la collisione, bisognerà consultare un'altra locazione di memoria e se la gestione avviene in Basic, il tutto si svolge con esasperante lentezza.

Con la routine proposta, invece, saremo addirittura in possesso di una nuova para-istruzione! Avete mai sentito parlare di una SYS XXX THEN?

Ebbene, la sintassi è proprio giusta (non avete letto male), e per accertarvene potete utilizzare il programma dimostrativo, che dissiperà sicuramente ogni dubbio.

Dopo la parola SYS andranno posti, separati da virgole, un paio di argomenti: il primo determina il numero dello sprite da verificare (da 0 a 7); il secondo, invece, determina il tipo di collisione che si vuole verificare: uno (1), se si controlla il testo e zero (0) se si guarda la collisione con un altro sprite.

In quest'ultimo caso se due sprite vengono in contatto tra loro (uno dei due deve essere specificato nell'istruzione) allora, oltre ad avere l'esecuzione delle istruzioni dopo THEN, la locazione 252 [PEEK(252)] conterrà il valore dell'altra o delle altre animazioni tra le quali si è verificato il contatto (espresso in notazione binaria).

L'articolo "Tutto sugli Sprite" (CCC N.35) è piuttosto soddisfacente per chiarire problemi relativi alle collisioni, ma vale la pena di ricordare che il registro che verifica i contatti, dopo essere stato letto, automaticamente si cancellerà e resterà a zero fino a quando non varierà la posizione di almeno un'animazione.

Il programma proseguirà regolarmente dopo l'istruzione THEN se la collisione richiesta si è verificata, altrimenti, come in un regolare IF... THEN, si passerà immediatamente alla successiva riga di programma.

```
20 DW$=CHR$(17)
100 X=21840:REM INDIRIZZO SUGGE
    RITO SU COMMODORE COMPUTER
    CLUB
110 PRINTCHR$(147)DW$DW$"COLLIS
    IONE SPRITE NERO COL TESTO
    :"
120 PRINTDW$"COLLISIONE SPRITE
    ROSSO COL TESTO :"
130 PRINTDW$"COLLISIONE TRA LE
    DUE SPRITES :"
140 PRINTCHR$(19):FOR I=1 TO 10
    :PRINTDW$;:NEXT:PRINT TAB(5
    )"COMMODORE 64"
150 FOR I=0 TO 63:POKE 832+I,25
    5:NEXT
160 POKE 2040,13:POKE 2041,13:P
    OKE 53269,3:POKE 53287,0:PO
    KE 53288,2
170 POKE 53249,130:POKE 53249,1
    30:POKE 53251,130
180 FOR I=0 TO 255 STEP 4:I1=25
    5-I
190 POKE 53248,I:POKE 53250,I1:
    PRINTCHR$(19);
200 SYSX,0,1 THEN PRINT TAB(36)
    DW$;DW$"SI":GOTO 220
210 PRINT TAB(36)DW$;DW$"NO"
220 SYSX,1,1 THEN PRINT TAB(36)
    DW$"SI":GOTO 240
230 PRINT TAB(36)DW$"NO"
240 SYSX,0,0 THEN PRINT TAB(36)
    DW$"SI":GOTO 260
250 PRINT TAB(36)DW$"NO"
260 FOR K=1 TO 200:NEXTK:NEXT:G
    OTO 180
```

```
1000 PRINTCHR$(147)"VERIFICA COL
     LISIONI"
1001 PRINT"[2 DOWN]SYS XXXXX, (P
     ARAMETRI) THEN"
1002 PRINT"[DOWN]NUMERO SPRITE (
     0-7)"
1003 PRINT"SPRITE/TESTO (0,1)"
1004 PRINT"[DOWN]I PARAMETRI VAN
     NO SEPARATI CON VIRGOLA. TU
     TTI SONO NECESSARI"
1005 PRINT"[DOWN]ESEGUE ISTRUZIO
     NE DOPO 'THEN' SE"
1006 PRINT"VERIFICA COLLISIONE"
1099 RETURN
1100 DATA 032,253,174,032,138,17
     3,032,247,183,165
1110 DATA 021,208,027,165,020,20
     1,008,016,021,133
1120 DATA 251,032,253,174,032,13
     8,173,032,247,183
1130 DATA 165,021,208,006,165,02
     0,201,002,048,003
1140 DATA 076,072,178,170,169,00
     1,164,251,240,004
1150 DATA 010,136,208,252,133,25
     1,189,030,208,133
1160 DATA 252,037,251,240,008,06
     9,252,133,252,169
1170 DATA 001,208,002,133,252,13
     3,097,076,043,169
1180 DATA -1,10255
```

Routine di pause (21920/21962)

Questa brevissima routine in linguaggio macchina non fa altro che sospendere l'esecuzione del programma per il tempo indicato (in decimi di secondo).

Alcuni si chiederanno che utilità abbia una routine di tal genere quando si può semplicemente utilizzare un ciclo For...Next. Ebbene, di vantaggi ve ne sono ben due:

- è possibile interrompere la pausa impostata premendo un qualsiasi tasto, e questa è la caratteristica meno importante.

- la caratteristica notevole del microprogramma presentato è rappresentata dalla precisione della pausa: infatti quest'ultima, oltre ad essere indicata in decimi di secondo, può variare da 1/10" a 65535/10" (quasi 2 ore) e gode di una precisione molto elevata soprattutto su tempi medio-lunghi: provate a cronometrarla: con una impostazione di 10 secondi l'errore è di 1/10": ne deriva un errore percentuale dello 0.017%

Potrete quindi usare la routine per regolare un orologio di un programma Basic e, perchè no?, con le opportune modifiche inserirla all'interno di un'altra routine in linguaggio macchina che visualizzi un orologio.

```
1000 PRINTCHR$(147)"QUESTA ROUTI
     NE GENERA UNA PAUSA DI UN"
1010 PRINT"NUMERO VOLUTO DI DECI
     MI DI SECONDO, LA"
1020 PRINT"PAUSA PUO' ESSERE INT
     ERROTTA PREMENDO"
1030 PRINT"UN TASTO QUALSIASI"
1040 RETURN
1050 DATA 032,253,174,032,138,17
     3,032,247,183,032
1060 DATA 228,255,208,028,160,09
     7,032,179,238,234
1070 DATA 234,136,208,248,160,00
     2,136,208,253,198
1080 DATA 020,208,232,166,021,24
     0,005,198,021,024
1090 DATA 144,223,096,-1,6336
```

```
10 REM DEMO ROUTINE DI PAUSE
20 X=21920:REM INDIRIZZO SUGGE
   RITO SU CCC
30 PRINT"[CLEAR]QUESTA E' UNA
   PAUSA DI 5 SECONDI, PER"
40 PRINT"INTERROMPERLA PUOI PR
   EMERE UN TASTO."
50 SYSX,50
60 PRINT"FINE"
```

Mappa della memoria di NUOVO SISTEMA

(Elenco delle routine pubblicate)

Il primo valore indica l'indirizzo di partenza (coincidente con la SYS da impartire), mentre, il secondo, l'ultima locazione contenente l'ultimo dato.

Il numero fra parentesi, invece, si riferisce al numero di C.C.C. in cui sono state pubblicate le routine stesse.

20000/20011 GoTo Calcolato (31)
20012/20049 GoSub Calcolato (31)
20050/20128 Interp A$ (31)
20129/20188 Cambia colore (31)
20189/20245 Scroll Carattere (31)
20246/20302 Cancella caratt. (31)
20303/20445 GoSub Label (32)
20446/20562 GoTo Label (32)
20563/20596 Restore linea (33)
20597/20682 Disk Tool (33)
20683/20775 Directory (33)
20776/20858 Scroll Flag (34)
20859/20914 Deek (34)
20915/20952 Doke (34)
20953/21106 Decim/Esadec (35)
21107/21156 Locate cursor (35)
21157/21260 Beep (35)

(Le routine di questo numero sono opera di Simone Bettola e Silvio Callegari)

Registrate il mio abbonamento annuale a Commodore Computer Club.

☐ Ho versato oggi stesso il canone di L. 35.000 a mezzo c/c postale n° 37952207 intestato a:
Systems Editoriale - V.le Famagosta, 75 - 20142 Milano

☐ Ho inviato oggi stesso assegno bancario n. ....
per l'importo di L. 35.000 intestato a Systems Editoriale
Si prega di scrivere il proprio nome e l'indirizzo completo in modo chiaro e leggibile. Inviare la fotocopia del bollettino di c/c postale.

Considerando che i numeri 1, 2 e 7 sono esauriti, vogliate inviarmi i numeri arretrati .... al prezzo di L. 5.000 cadauno per richieste fino a 4 numeri, o di L. 4.000 cadauno per richieste oltre i 4 numeri arretrati, e perciò per un totale di L. .... Sono a conoscenza che i fascicoli suddetti non saranno inviati in contrassegno e, pertanto, ho provveduto oggi stesso a versare il canone di L. .... a mezzo c/c postale n. 37952207 intestato a:
Systems Editoriale - V.le Famagosta, 75 - 20142 Milano

PICCOLI ANNUNCI ....

CERCO/OFFRO CONSULENZA ....

INVIARE IN BUSTA CHIUSA E AFFRANCANDO SECONDO LE TARIFFE VIGENTI A:

COMMODORE COMPUTER CLUB

V.le Famagosta, 75
20142 Milano

Nome .... Cognome ....

Via .... N° .... CAP. .... Città ....

Telefono .... Orario ....

INVIARE TUTTA LA PAGINA ANCHE SE SI UTILIZZA UNA SOLA SCHEDA

PRESENTA

Software Club

C64/C128

Cover (10 K)

Moscow Summit (32 K)

Poing (29 K)

Derby (21 K)

Around the world (39 K)

Leasing (24 K)

Vc 20

Cover (2 K)

Crash (3 K)

Protector (2 K)

C16/+4

Cover (3 K)

The wall (5 K)

Derby (12 K)

Space 1999 (8 K)

Spectrum

Village II (11 K)

Visitors III (16 K)

Base Alpha (18 K)

OM (15 K)

MSX

Tape data base (9 K)

Slot machine (10 K)

In edicola

Lire 8.000

Commodore Club # 13 - Dir. Resp.: Agostina Ronchetti - Edizioni: Systems Editoriale srl - V.le Famagosta 75 - 20142 Milano - Reg. Trib. Milano n. 104 del 25/2/84 - Distr.: MePe - Prodotto in Italia